

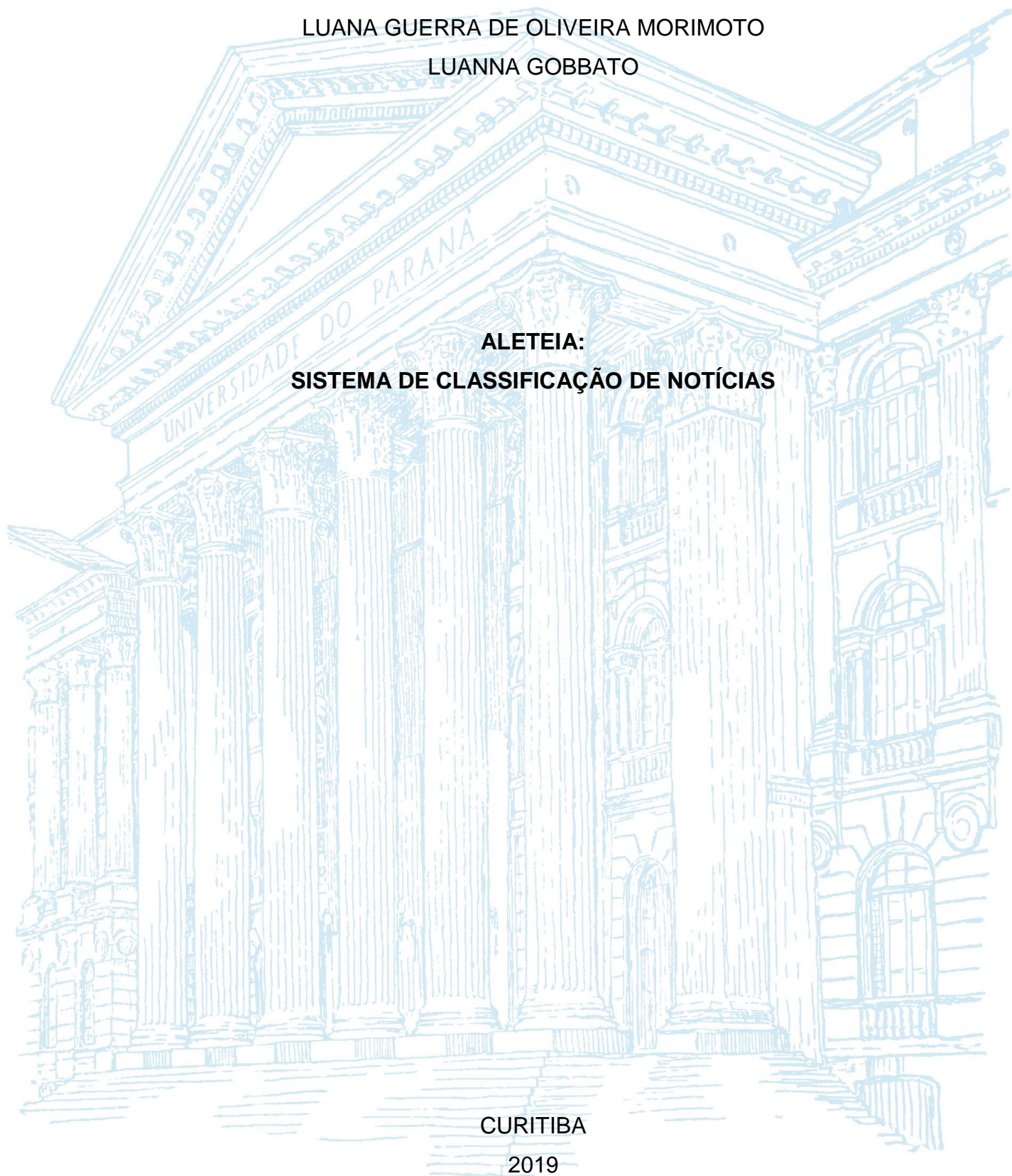
UNIVERSIDADE FEDERAL DO PARANÁ

HARUAN MOSSATO JUSTINO  
LUANA GUERRA DE OLIVEIRA MORIMOTO  
LUANNA GOBBATO

**ALETEIA:  
SISTEMA DE CLASSIFICAÇÃO DE NOTÍCIAS**

CURITIBA

2019



HARUAN MOSSATO JUSTINO  
LUANA GUERRA DE OLIVEIRA MORIMOTO  
LUANNA GOBBATO

**ALETEIA:**  
**SISTEMA DE CLASSIFICAÇÃO DE NOTÍCIAS**

Trabalho de conclusão de curso apresentado ao curso de Tecnologia em Análise e Desenvolvimento de Sistemas da Universidade Federal do Paraná, como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientadora: Profa. Dra. Rafaela Mantovani Fontana

CURITIBA  
2019

## TERMO DE APROVAÇÃO

HARUAN MOSSATO JUSTINO  
LUANA GUERRA DE OLIVEIRA MORIMOTO  
LUANNA GOBBATO

### ALETEIA: SISTEMA DE CLASSIFICAÇÃO DE NOTÍCIAS

Trabalho de Conclusão de Curso apresentado ao curso Tecnólogo em Análise e Desenvolvimento de Sistemas da Universidade Federal do Paraná, como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, pela seguinte banca examinadora:



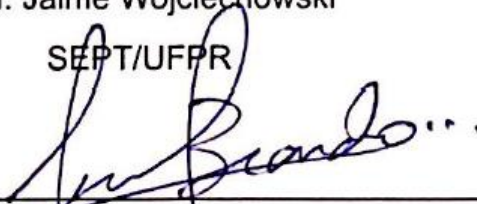
---

Prof<sup>a</sup>. Rafaela Mantovani Fontana  
Orientadora – SEPT/UFPR



---

Prof. Jaime Wojciechowski  
SEPT/UFPR



---

Prof. Evandro Luiz Brandão  
SEPT/UFPR

Curitiba, 24 de junho de 2019.

*“Se você não for cuidadoso,  
os jornais farão você odiar os oprimidos  
e amar os opressores.”*

*Malcolm X.*

## **AGRADECIMENTOS**

Agradecemos às progenitoras Rosana Ferreira, Maria Silzi Mossato, Cynara de Souza Guerra e Almerinda de Souza, pois sem seu apoio e amor incondicional, este trabalho não teria sido concluído com tamanho esmero. E aos familiares que estiveram sempre ao nosso lado nesta empreitada, nos auxiliando sempre que necessário.

Também aos incentivadores Jaqueline Martins Niedzwiedz, Geovani Janiski, Jefferson dos Santos Ambrósio, Alan Peterson Carvalho Silva, Gabriel Peixoto e Felipe de Melo Becel, que por inúmeras vezes deixaram de lado seus interesses pessoais para nos auxiliar em nossas dificuldades e oferecer um ombro amigo.

De igual importância foram nossos professores Razer Montañó e Roberto Tadeu Raittz, por nos ajudarem com nossas várias dúvidas técnicas sempre que precisávamos, e nossa orientadora Rafaela Mantovani Fontana, que além da grande ajuda durante todo o processo de desenvolvimento deste trabalho, sempre se preocupou com o nosso equilíbrio entre a vida acadêmica e pessoal.

Por último e não menos importante, às responsáveis pela nossa sanidade mental e por não nos deixarem ceder, as maravilhosas Paçoca, Nori, Azuki, Princesa e Laurinha.

## RESUMO

Este trabalho de conclusão de curso visa apresentar uma ferramenta para detectar as notícias falsas tão amplamente disseminadas na atualidade, principalmente com o estabelecimento das redes sociais. A quantidade alarmante de notícias enganosas, juntamente com o crescente número de usuários com o poder de compartilhá-las sem o devido conhecimento para verificar suas fontes e veracidade, levaram a população mundial à uma série de tomadas de decisões errôneas e que não refletem suas reais intenções, seja no âmbito político, social ou até mesmo da saúde. Órgãos jornalísticos foram criados com o intuito de conter esta epidemia, porém com pouco pessoal e investimento é quase impossível circunvir a quantidade de notícias falsas que surgem a cada momento. Por outro lado, nas últimas décadas foi possível acompanhar o surgimento e desenvolvimento de tecnologias baseadas em Inteligência Artificial, que ampliam o alcance das tarefas realizadas por entidades computacionais, como é o caso de algo tão abrangente e disforme como a detecção automática de notícias falsas. A aplicação Aleteia traz uma nova abordagem para a tratativa deste problema e uma possível solução para levar o conhecimento às mãos do usuário final. Por intermédio de uma extensão para o navegador Google Chrome e desenvolvida na linguagem de programação Python sob o conceito de Aprendizado de Máquina, a aplicação aprende os vícios mais comumente utilizados nos dois tipos de informação (falsa ou verdadeira) por meio de processamento de linguagem natural, podendo então classificar uma notícia fornecida por este usuário, intuindo-o com relação à autenticidade do texto.

Palavras-chave: *Fake News*. Notícias falsas. Extensão. Google Chrome. Aleteia. Inteligência Artificial. Aprendizado de Máquina. Imprensa. *Bag-of-Words*.

## **ABSTRACT**

This undergraduate thesis aims to present a tool with the objective of detecting the fake news widely spread in the later years, especially after the establishment of social networks. The alarming amount of misleading news, coupled with the exponentially increasing number of users with the power to share them without the proper knowledge to verify their sources and truthfulness, have led the world population to a series of erroneous decision-making that do not reflect their real intentions, whether in the political, social or even health sphere. Organizations were created with the intention of containing this epidemic, but with little personnel and investment it is almost impossible to circumvent the amount of fake news that keep appearing at any given moment. On the other hand, in the last decades it was possible to follow the development of Artificial Intelligence concept, extending the reach of tasks performed by computational entities, now ready to face much more complex situations such something as comprehensive and misshapen as the automatic detection of false news. Aleteia brings a new approach to the treatment of this problem and a possible solution to bring knowledge into the hands of the end user. Through a Google Chrome extension and developed under the Python programming language and the Machine Learning concept, the application learns the most common language vices of the two types of information (true or fake) through Natural Language Processing, being able to classify a news text provided by this user, intuiting them about the authenticity of this text.

**Keywords:** Fake News. False news. Extension. Google Chrome. Aleteia. Artificial intelligence. Machine Learning. Press.

## LISTA DE FIGURAS

FIGURA 1 - EXEMPLO DE APRENDIZADO DE MÁQUINA.....	20
FIGURA 2 - EXEMPLO DE SVM.....	22
FIGURA 3 - TREINAMENTO DO MODELO DE CLASSIFICAÇÃO .....	23
FIGURA 4 - APLICAÇÃO CONTEINERIZADA E MÁQUINA VIRTUAL .....	26
FIGURA 5 - METODOLOGIA SCRUM.....	36
FIGURA 6 - UTILIZAÇÃO DA FERRAMENTA TRELLO .....	37
FIGURA 7 - UTILIZAÇÃO DO GOOGLE AGENDA.....	37
FIGURA 8 - ESTRUTURA DA APLICAÇÃO .....	47
FIGURA 9 - FUNCIONAMENTO DA APLICAÇÃO.....	48
FIGURA 10 - EXTENSÃO: ÍCONE E INSTRUÇÕES .....	49
FIGURA 11 - EXTENSÃO: ENVIANDO O TEXTO PARA VERIFICAÇÃO.....	50
FIGURA 12 - EXTENSÃO: RESULTADO NOTÍCIA FALSA.....	50
FIGURA 13 - EXTENSÃO: RESULTADO NOTÍCIA VERDADEIRA.....	51
FIGURA 14 - ADMINISTRADOR: HOME PAGE .....	51
FIGURA 15 - ADMINISTRADOR: CLASSIFICAÇÃO MANUAL .....	52
FIGURA 16 - ADMINISTRADOR: ADICIONAR NOTÍCIA .....	53
FIGURA 17 - ADMINISTRADOR: ALTERAR NOTÍCIAS .....	54
FIGURA 18 - ADMINISTRADOR: REQUISIÇÕES DE CLASSIFICAÇÃO .....	55

## LISTA DE TABELAS

TABELA 1 - ESTATÍSTICAS SCIKIT-LEARN .....	32
TABELA 2 - RESUMO DAS SPRINTS.....	39
TABELA 3 - OPÇÕES DE TECNOLOGIA PARA O NÚCLEO DA APLICAÇÃO.....	43
TABELA 4 - APLICAÇÕES DE SUPORTE .....	44
TABELA 5 - MATRIZ CONFUSÃO.....	56
TABELA 6 - ESTATÍSTICAS DO MODELO.....	56

## LISTA DE ABREVIATURAS OU SIGLAS

HTTP	- Hyper Text Transfer Protocol
TCP	- Transmission Control Protocol
VM	- Virtual Machine
NLP	- Natural Language Processing
SVM	- Support Vector Machine
BOW	- Bag-of-Words
API	- Application Programming Interface
NLTK	- Natural Language Tool-kit
DOM	- Document Object Model
JSON	- JavaScript Object Notation
XML	- Extensible Markup Language
REST	- Representational State Transfer
HTML	- Hypertext Markup Language
CSS	- Cascading Style Sheets
UML	- Unified Modeling Language

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	<b>13</b>
1.1 OBJETIVO GERAL .....	15
1.2 OBJETIVOS ESPECÍFICOS .....	16
1.3 JUSTIFICATIVA .....	16
1.4 ESTRUTURA DO DOCUMENTO.....	17
<b>2 FUNDAMENTAÇÃO TEÓRICA</b> .....	<b>19</b>
2.1 CONCEITOS .....	19
2.1.1 APRENDIZADO DE MÁQUINA.....	19
2.1.2 SUPPORT VECTOR MACHINE.....	21
2.1.3 TREINAMENTO DOS MODELOS DE CLASSIFICAÇÃO .....	22
2.1.4 PROCESSAMENTO DE LINGUAGEM NATURAL .....	23
2.1.5 BAG OF WORDS .....	24
2.1.6 WEB CRAWLER E WEB SCRAPING .....	25
2.1.7 CONTEINERIZAÇÃO .....	26
2.2 FERRAMENTAS .....	27
2.2.1 API CHROME.....	27
2.2.2 TOR.....	27
2.2.3 PRIVOXY E HAPROXY.....	28
2.2.4 DOCKER.....	28
2.3 LINGUAGENS DE PROGRAMAÇÃO .....	29
2.3.1 PYTHON .....	29
2.3.2 JAVASCRIPT .....	30
2.4 BIBLIOTECAS.....	31
2.4.1 SCIKIT-LEARN.....	31
2.4.2 NLTK – NATURAL LANGUAGE TOOL-KIT .....	32
2.4.3 SCRAPY FRAMEWORK.....	32
2.4.4 DJANGO PROJECT E DJANGO REST FRAMEWORK .....	33
2.4.5 JQUERY .....	33
<b>3 MATERIAIS E MÉTODOS</b> .....	<b>35</b>
3.1 SCRUM .....	35
3.1.1 CRONOGRAMA E GERENCIAMENTO DE TEMPO .....	36
3.1.2 SPRINTS.....	38

3.1.2.1 SPRINT 1 .....	39
3.1.2.2 SPRINT 2 .....	40
3.1.2.3 SPRINT 3 .....	40
3.1.2.4 SPRINT 4 .....	40
3.1.2.5 SPRINT 5 .....	40
3.1.2.6 SPRINT 6 .....	41
3.1.2.7 SPRINT 7 .....	41
3.1.2.8 SPRINT 8 .....	41
3.1.2.9 SPRINT 9 .....	41
3.2 UML – LINGUAGEM DE MODELAGEM UNIFICADA.....	42
3.3 TECNOLOGIAS UTILIZADAS.....	42
3.3.1 GITHUB.....	44
<b>4 RESULTADOS.....</b>	<b>46</b>
4.1 O SISTEMA.....	46
4.2 O PROCESSO .....	47
4.3 COMO UTILIZAR O SISTEMA.....	49
4.4 TESTES E RESULTADOS OBTIDOS.....	55
<b>5 CONSIDERAÇÕES FINAIS .....</b>	<b>58</b>
5.1 RECOMENDAÇÃO DE TRABALHOS.....	58
<b>REFERÊNCIAS.....</b>	<b>60</b>
<b>APÊNDICE A - HISTÓRIAS DE USUÁRIOS.....</b>	<b>64</b>
<b>APÊNDICE B - ESPECIFICAÇÕES DE CASOS DE USO .....</b>	<b>67</b>
<b>APÊNDICE C - DIAGRAMA DE CASOS DE USO .....</b>	<b>90</b>
<b>APÊNDICE D - DIAGRAMA DE IMPLEMENTAÇÃO .....</b>	<b>91</b>
<b>APÊNDICE E - DIAGRAMA DE CLASSE: MODEL .....</b>	<b>92</b>
<b>APÊNDICE F - DIAGRAMA DE CLASSE: PACOTE ALETEIA_CLASSIFICATOR</b>	<b>93</b>
<b>APÊNDICE G - DIAGRAMA DE CLASSE: API.....</b>	<b>94</b>
<b>APÊNDICE H - DIAGRAMA DE CLASSE: EXTENSÃO .....</b>	<b>95</b>
<b>APÊNDICE J - DIAGRAMA DE CLASSE: SERVICE CORE .....</b>	<b>97</b>
<b>APÊNDICE K - DIAGRAMA DE CLASSE: NEWSCRAWLER.....</b>	<b>98</b>
<b>APÊNDICE L - DIAGRAMA DE CLASSE: SERVICES .....</b>	<b>99</b>
<b>APÊNDICE M - DIAGRAMA DE ATIVIDADES: ALETEIA.....</b>	<b>100</b>
<b>APÊNDICE N - DIAGRAMA DE ATIVIDADES: EXTENSÃO .....</b>	<b>101</b>
<b>APÊNDICE O - DIAGRAMA DE SEQUÊNCIA DE ANÁLISE: EXTENSÃO.....</b>	<b>102</b>

<b>APÊNDICE P - DIAGRAMA DE SEQUÊNCIA: LOGIN.....</b>	<b>103</b>
<b>APÊNDICE Q - DIAGRAMA DE SEQUÊNCIA: BUSCAR NOTÍCIA.....</b>	<b>104</b>
<b>APÊNDICE R - DIAGRAMA DE SEQUÊNCIA: CRAWLEAR NOTÍCIAS .....</b>	<b>105</b>
<b>APÊNDICE S - DIAGRAMA DE SEQUÊNCIA: TREINAR MODELO DE CLASSIFICAÇÃO.....</b>	<b>106</b>
<b>APÊNDICE T - DIAGRAMA DE SEQUÊNCIA: VERIFICAR VERACIDADE DA NOTÍCIAS / SOLICITAR CLASSIFICAÇÃO / CLASSIFICAR NOTÍCIA - 1.....</b>	<b>107</b>
<b>APÊNDICE T - DIAGRAMA DE SEQUÊNCIA: VERIFICAR VERACIDADE DA NOTÍCIAS / SOLICITAR CLASSIFICAÇÃO / CLASSIFICAR NOTÍCIA - 2.....</b>	<b>108</b>
<b>APÊNDICE U - MODELO LÓGICO DE DADOS .....</b>	<b>109</b>
<b>APÊNDICE V - DIAGRAMA DE ARQUITETURA.....</b>	<b>110</b>

## 1 INTRODUÇÃO

É epidêmico o advento do compartilhamento das notícias falsas, especialmente nos últimos anos. Com a crescente popularização do acesso à internet no Brasil e no Mundo, cresce também a circulação de informações falsas, as chamadas “*fake news*”, mergulhando as pessoas em um mar turvo de desinformação, onde não é mais possível separar facilmente o real do produzido.

A eleição para presidente dos Estados Unidos de 2016 mostrou o impacto direto que essas notícias falsas podem ter sobre a população, como analisado por Tardáguila (2018), redatora do portal verificador de fatos Lupa. As campanhas dos principais candidatos Donald Trump e Hillary Clinton foram permeadas por notícias falsas e teorias da conspiração, como por exemplo o “*Pizzagate*”, uma teoria fantasiosa que colocava a candidata Hillary no centro de uma suposta rede de pedofilia e tráfico infantil que nunca existiu. Ou então a notícia mais compartilhada das eleições, sobre a preferência do Papa Francisco pelo candidato Trump (EVON, 2016), uma matéria proveniente de um site satírico, mas que provocou tantos compartilhamentos, que precisou ser desmentida por outra organização verificadora de fatos, a Snopes (2016). Ainda sobre estas eleições, um estudo realizado pelo site BuzzFeed selecionou as 20 notícias mais importantes provenientes de jornais de confiança, e confrontou o seu desempenho com relação às 20 notícias falsas mais lidas pelos usuários do Facebook ao longo dos meses de fevereiro até o dia da eleição, em agosto. Esta análise mostrou que, próximo à data da eleição, as notícias falsas chegaram a superar o alcance de público em praticamente um milhão e meio de compartilhamentos em relação às notícias genuínas, ilustrando a preocupante situação em que se encontram as informações consumidas pelo público geral (SILVERMAN, 2016).

A área da saúde também é afetada por esse fenômeno, gerando desassossego entre os profissionais do setor. A epidemiologista e chefe de estratégia de combate à doença da Organização Mundial da Saúde (OMS), culpa o compartilhamento de notícias falsas, tais como a falta de eficácia da vacina fracionada e a sua possível substituição pelo própolis, pela baixa porcentagem de brasileiros vacinados contra a febre amarela, que consta em 55% da população, quando o ideal deveria estar em 80% (TIMOTEO, 2018).

Preocupados com o efeito desta “epidemia”, muitos grandes nomes da Imprensa resolveram criar seus próprios setores de checagem de fatos, com o objetivo de encontrar informações falsas que estejam sendo compartilhadas em larga escala, e desmascará-las por meio de dados reais, verificações e fontes. A *Annenberg Public Police Center* foi uma das pioneiras, com a criação do “*Factcheck.org*” em dezembro de 2003, seguida pelo gigante “*Politifact*”, nascido do tradicional jornal da Flórida Tampa Bay Times. Outras organizações também nasceram desta necessidade, como as agências “Aos Fatos” e “Lupa” no Brasil, além dos nomes “Snopes”, dos Estados Unidos, e “Chequeado”, baseado na Argentina. Muitas destas organizações foram criadas à luz de grandes eleições com o objetivo de verificar as informações proferidas pelos candidatos nos debates, principalmente presidenciais, provendo à população com um comparativo e uma verificação de fatos do que estaria sendo afirmado pelo candidato.

Algumas organizações deixaram de existir ao fim destas eleições, e outras deram continuidade a este trabalho de verificação, ampliando suas áreas de atuação para além da política e abrangendo também as áreas sociais e da saúde. Porém todas essas organizações compartilham das mesmas dificuldades: falta de pessoal e de incentivo monetário, e excesso de material a ser analisado. Conforme dados levantados durante o 5º Seminário Global de Checagem de Fatos (*Anual Global Fact-Checking Summit*), 64,3% das organizações existentes com esse propósito são Organizações Não Governamentais, sem fonte fixa de lucro (FUNKE, 2018). Outro dado observado consiste na quantidade de funcionários trabalhando nestas organizações, onde a maioria (15 de 42) funciona com menos de 5 pessoas dedicadas em tempo integral, isto é, uma média de 6 funcionários por organização. Com um elevado e ascendente número de notícias falsas sendo produzidos a cada instante, é de se esperar que tais organizações, levando em conta sua estrutura financeira e pessoal, enfrentem um grande desafio em verificar todos os fatos. Um estudo sobre a rede social Twitter apresentado pela Universidade de Warwick mostrou que, rumores falsos criados pelos pesquisadores em prol desta análise levavam de 15 a 20 horas para serem desbancados, tempo mais do que suficiente para alcançar uma larga escala de leitores até que seja identificado como falso (ZUBIAGA et al, 2016).

Já existem iniciativas de aplicações e robôs que utilizam os conceitos de inteligência artificial criados com o propósito de fazer verificações iniciais que possam indicar possíveis informações erradas ou falsas em uma certa afirmação, como o

*Chequeabot*, da agência argentina de verificação de fatos “Chequeado”, ainda em fase de testes, e utilizado apenas pelos próprios funcionários da agência. O portal Aos Fatos também desenvolveu uma ferramenta com o nome de “Fátima”, utilizada através do aplicativo de conversação do *Facebook*, o *Messenger*, onde é possível informar uma notícia, ou o link para esta, e a ferramenta informa se a notícia em questão já foi analisada pelo portal e qual foi o resultado da análise. A própria inspiração para o sistema Aleteia, o Estudo de Detecção de Notícias Falsas para o Português do Programa Institucional de Bolsas de Iniciação Científica, também foi disponibilizado para o público na forma de uma interface em uma página da web onde o usuário insere o corpo da notícia, e o sistema por sua vez analisa as informações, e retorna ao usuário se a notícia é verdadeira ou não (MONTEIRO, 2018).

O crescimento exponencial do chamado “conteúdo criado por usuários”, que se apoia no vasto número de cidadãos com acesso a um dispositivo que grave vídeos ou capture fotos (em sua maioria, o smartphone), fez com que a análise de imagens e textos de caráter explícito, violento ou perturbador façam parte do dia a dia de profissionais da área do jornalismo, incluindo os que fazem parte das organizações verificadoras de fatos acima mencionadas, Direitos Humanos e entre outras, responsáveis pela filtragem e veiculação de conteúdo informativo. Por este motivo, sintomas comumente associados ao Transtorno de estresse pós-traumático agora são amplamente observados neste tipo de ambiente de trabalho. Uma pesquisa realizada pela organização *First Draft* concluiu que mais da metade destes profissionais (52% dos jornalistas e 57% dos profissionais da área de Direitos Humanos) está em contato com este tipo de conteúdo diariamente, e 40% afirmam que a visualização destes conteúdos tem um impacto direto e negativo em sua vida pessoal, levando-os inclusive a desenvolver mecanismos para lidar com estes sentimentos, tais como o abuso de drogas e álcool, compulsões alimentares, entre outros (DUBBERLEY; GRIFFIN, 2015).

## 1.1 OBJETIVO GERAL

Este trabalho possui o objetivo de desenvolver uma aplicação que, disponível para o usuário na forma de uma extensão para navegador de internet, possa receber deste usuário uma notícia, e após processar o texto contido nesta, seja capaz de

classifica-la com relação à sua veracidade, e intuir o usuário e leitor com relação a autenticidade desta.

## 1.2 OBJETIVOS ESPECÍFICOS

- Desenvolver um módulo de classificação de notícias que, ao utilizar ferramentas de aprendizado de máquina e modelos de processamento de linguagem natural, seja capaz de classificar um dado texto de acordo com o nível de veracidade e confiabilidade das suas informações;
- Criar um módulo de treinamento que, utilizando ferramentas de coleta de informação, archive em um banco de dados notícias que já foram previamente investigadas por órgãos verificadores de fatos, e que serão utilizadas para a comparação com os textos enviados pelo usuário, com o objetivo de treinar o módulo de classificação.
- Disponibilizar uma extensão a ser utilizada no navegador para que o usuário selecione a informação a ser verificada, e a extensão, por sua vez, envie esta informação aos módulos através de uma API, que após processados, retornarão um resultado que será apresentado visualmente ao usuário.

## 1.3 JUSTIFICATIVA

Aleteia, a forma divina da verdade na mitologia grega, segundo uma das fábulas de Ésope, dá nome a este trabalho na forma de uma aplicação que possibilita o leitor de uma certa notícia ter em suas mãos uma ferramenta que o auxilie a discernir o correto do falso, desta maneira quebrando as correntes de disseminação de “desinformação”.

Com a popularização das redes sociais, o leitor passou a ter o poder da divulgação da informação, e por vezes até da criação de conteúdo, os quais não recebem um devido estudo ou filtro como o fariam se estivessem em mãos de profissionais como jornalistas ou escritores.

O gigante Facebook atualmente conta com 127 milhões de usuários ativos no Brasil, segundo pesquisa realizada no primeiro trimestre de 2018. Isso significa que

mais da metade (65%) da população brasileira com 10 anos ou mais possui acesso à esta plataforma, número que chega na casa dos 2,2 bilhões quando mensurado globalmente. Já o aplicativo de mensagens WhatsApp conta com 120 milhões de usuários no Brasil e 1,5 bilhão no mundo. Estes números somados à quantidade não suficiente de organizações focadas em filtrar e verificar essas informações, e a sua dificuldade em acompanhar o volume de notícias geradas a cada instante, conforme apresentado nos capítulos anteriores, dá forma à necessidade de colocar o poder e o conhecimento nas mãos do usuário, capacitando-o para que ele mesmo possa verificar as fontes e fatos contidos em uma certa notícia antes de publicá-la ou compartilhá-la.

Uma ferramenta utilizada diretamente pelo leitor de uma certa notícia diminui a dependência das organizações verificadoras de fatos para que seja checada a veracidade da notícia em questão, simplificando e agilizando o processo. Com esta solução em mente, foi criada a Aleteia, que por meio de uma extensão para o navegador da Internet Google Chrome, classifica a notícia selecionada pelo usuário em níveis de confiabilidade, retornando para o usuário esta informação de forma instantânea.

#### 1.4 ESTRUTURA DO DOCUMENTO

Este documento objetiva detalhar todo o processo de desenvolvimento do sistema, assim como as ferramentas usadas para tal e as fundamentações e motivos para a sua elaboração, conforme explicado a seguir.

O atual capítulo, Capítulo 1, demonstrou os motivos pelos quais o sistema foi desenvolvido, elaborou sua base e fundamentos para entendimento do mesmo, e reuniu de informações sobre o assunto ao qual permeia o sistema, as notícias falsas.

O Capítulo 2 aborda a Fundamentação Teórica do sistema, apresentando os recursos que foram utilizados no seu desenvolvimento, entre linguagens de programação, ferramentas e bibliotecas que foram úteis durante a sua elaboração, além de conceitos e tecnologias aplicados, detalhando o funcionamento e utilidade de cada um deles.

Já o Capítulo 3, mostra como estas ferramentas foram utilizadas na prática, explicitando sua metodologia e cronograma de desenvolvimento, detalhando a

administração de tempo da equipe e como foram gerenciadas as tarefas ao longo da produção deste trabalho.

O Capítulo 4 traz os resultados obtidos ao aplicar as metodologias do Capítulo 3 nas ferramentas apresentadas no Capítulo 2: o sistema em funcionamento, suas características e resultados. Como a utilização de todas as ferramentas e bibliotecas culminou nos módulos funcionais, que são o centro do sistema, e de que maneira isto é apresentado ao usuário em suas diversas formas.

No Capítulo 5 se dão as Considerações Finais, onde são apresentados os resultados obtidos e de que maneira isto reflete no alcance do objetivo proposto logo no início deste documento, além dos planos futuros para continuação e melhoria do projeto.

Nos apêndices deste documento constam as Histórias de Usuário (Apêndice A), Especificações de Caso de Uso (Apêndice B), Diagrama de Casos de Uso (Apêndice C), Diagrama de Implementação (Apêndice D), Diagramas de Classe – Model (Apêndice E), Pacote Aleteia\_Classificador (Apêndice F), API (Apêndice G), Extensão (Apêndice H), Service Admin (Apêndice I), Service Core (Apêndice J), Newscrawler (Apêndice K) e Services (Apêndice L), Diagramas de Atividades – Aleteia (Apêndice M) e Extensão (Apêndice N), Diagramas de Sequência – Login (Apêndice P), Buscar Notícia (Apêndice Q), Crawler Notícias (Apêndice R), Treinar Modelo de Classificação (Apêndice S) e Verificar Veracidade da Notícia (Apêndice T), Modelo Lógico de Dados (Apêndice U) e o Diagrama de Arquitetura (Apêndice V).

## 2 FUNDAMENTAÇÃO TEÓRICA

Visualizada pelo usuário como uma extensão do navegador de internet Google Chrome, a aplicação Aleteia utiliza o conceito de Aprendizado de Máquina, desenvolvida na linguagem de programação Python e previamente treinada com as informações necessárias por meio de uma máquina de vetor de suporte para processar os dados enviados, comunicando-os tanto no recebimento quanto no envio através de uma API.

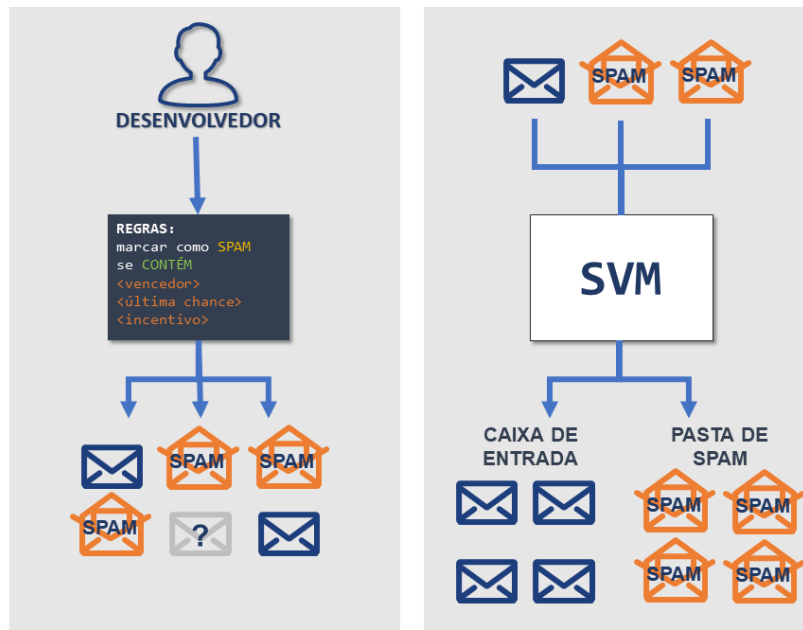
A seguir são apresentados os conceitos utilizados na concepção da aplicação, principalmente permeando a questão do Aprendizado de Máquina, além das ferramentas que auxiliaram em todo o processo de desenvolvimento, e as linguagens de programação e suas respectivas bibliotecas utilizadas.

### 2.1 CONCEITOS

#### 2.1.1 APRENDIZADO DE MÁQUINA

O Aprendizado de Máquina ou *Machine Learning*, é a prática de ensinar um software a executar uma tarefa, sem que necessariamente sejam utilizados comandos fixos para tal. Para Raschka e Mirjalili (2017), o principal objetivo do Aprendizado de Máquina é gerar um modelo através de dados classificados que permita “prever o futuro”, ou fazer previsões sobre dados ainda não analisados. Abaixo é demonstrado o exemplo de uma tarefa de classificação de e-mails, realizada por um algoritmo comum na esquerda e através de Aprendizado de Máquina na direita (FIGURA 1).

FIGURA 1 - EXEMPLO DE APRENDIZADO DE MÁQUINA



FONTE: Adaptado de TENSORFLOW.ORG (2019)

Existem quatro tipos de Aprendizado de Máquina: Supervisionado, Semi-supervisionado, Não-Supervisionado e Aprendizado por Reforço. No caso dos aprendizados supervisionados e semi-supervisionados, são fornecidos ao algoritmo os dados de entrada juntamente com os dados esperados de saída, para que, através de iteração otimizada, o algoritmo desenvolva a função para a classificação de novos dados. Já o aprendizado não-supervisionado utiliza apenas dados de entrada, localizando características similares entre eles, e agrupando-os conforme, sendo comumente utilizado para estimativas e no campo da estatística. E por fim, há o Aprendizado por Reforço, onde o algoritmo não assume nenhum tipo de modelo matemático, desta maneira sendo adequado para a utilização sobre dados inexatos, que não podem ser mensurados e consequentemente agrupados em vetores para posterior análise.

O Aprendizado de Máquina Supervisionado consiste em possuir previamente os resultados desejados para um conjunto de dados de entrada, também chamada de tabela de atributos e valores. É a partir dessa informação que será formulada uma hipótese sobre o que está sendo aprendido. O próprio algoritmo de aprendizagem é continuamente ajustado através da supervisão da discrepância entre as saídas produzidas em relação às saídas esperadas, usando essa diferença no procedimento de ajuste. O sistema é considerado "treinado" quando esta discrepância está dentro

de uma faixa de valores aceitável, apresentando resultados de eficácia altos para o objetivo para o qual foi proposto (SILVA et al, 2017).

Segundo Bowles (2015), a extração eficiente de dados está mudando o perfil de negócios atual e demandando novas habilidades de programação. Analistas de mercado estimam que a necessidade de profissionais com proficiência em Aprendizado de Máquina excederá o pessoal capacitado em 140 a 190 mil ainda nesta década.

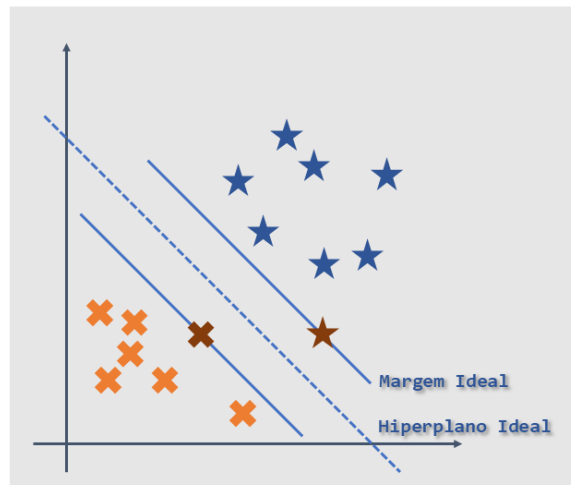
### 2.1.2 SUPPORT VECTOR MACHINE

As Máquinas de Vetores de Suporte ou SVMs são modelos de Aprendizagem de Máquina Supervisionada para problemas de classificação em dois grupos distintos que, ao ser mapeados em uma superfície de alta dimensionalidade, podem ser separados por um hiperplano linear, maximizando a separação destes dois grupos com o mínimo de erro (CORTES; VAPNIK, 1995).

Associados a algoritmos de aprendizagem, as SVM podem analisar dados para classificação e análise de regressão destes. A análise de regressão é um conjunto de processos estatísticos que estima o relacionamento entre variáveis dependentes e variáveis independentes, através da expectativa condicional da primeira sobre a segunda, isto é, um valor médio da variável dependente tratando as variáveis independentes não-fixas.

Um algoritmo de treinamento do tipo SVM recebe um conjunto de exemplos previamente classificados em duas categorias, e a partir desta informação gera uma representação na forma de pontos em um dado espaço, com o objetivo de mapear estes pontos agrupando-os em suas categorias, e que estes dois conjuntos de pontos estejam separados pelo maior espaço possível. A SVM faz essa separação através de um hiperplano colocado no dado espaço que contém os conjuntos de pontos. A SVM insere múltiplos hiperplanos no espaço, e calcula o que possui o maior espaço entre dois grupos de pontos, a chamada “margem” (FIGURA 2):

FIGURA 2 - EXEMPLO DE SVM



FONTE: Adaptado de CORTES; VAPNIK (1995).

Há também casos em que os dados não são passíveis de serem separados linearmente, onde aplica-se o “Truque Kernel”. Através de funções e cálculos avançados, a SVM analisa o grupo de dados através de uma perspectiva com uma dimensão adicional, encontrando outra maneira de ordenar os dados.

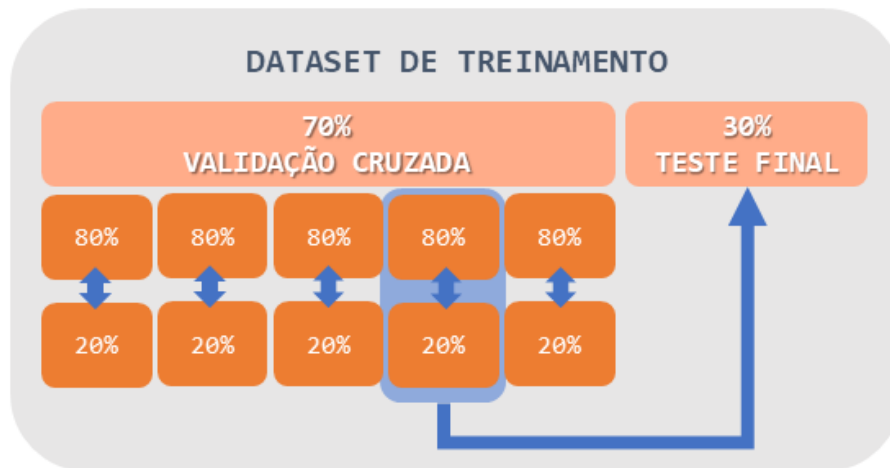
Como qualquer modelo de aprendizagem, a SVM também possui seus prós e contras, tendo margens claras de separação e grande eficiência em utilização de memória entre seus pontos positivos, porém apresentando baixa performance em casos com um grande conjunto de dados a serem analisados, devido ao tempo de treinamento requerido, além de demonstrar dificuldades em classificar dados muito “sujos”, como nos casos de classes que se sobrepõem.

### 2.1.3 TREINAMENTO DOS MODELOS DE CLASSIFICAÇÃO

A biblioteca utilizada para o Aprendizado de Máquina trabalha com um treinamento exaustivo dos modelos de classificação, para que seja produzido o modelo com maior acurácia e para que seja evitado o sobre ajuste. Sobre Ajuste ou *Overfitting* consiste em um modelo que foi treinado com uma especificidade muito grande sobre apenas um grupo de dados. Isto significa que este modelo terá uma acurácia muito grande sobre este grupo de dados específico, porém terá um desempenho baixo em dados que não foram utilizados no treinamento ou novos dados

de entrada. O treinamento é realizado da seguinte maneira: a totalidade dos dados de entrada já classificados é dividida em dois blocos aleatórios na proporção de 70 e 30 por cento, sendo o primeiro bloco utilizado para treinamento através de validação cruzada, e o segundo para que seja realizado um teste final do modelo. O bloco de validação cruzada, por sua vez, é dividido em outras cinco partes, onde cada uma destas partes é dividida em 80 e 20 por cento para que sejam treinadas entre si através da técnica de validação cruzada. Destes cinco modelos, o que apresentar maior acurácia após a validação, será colocado em teste contra os 30 por cento separados no início do processo de treinamento (FIGURA 3):

FIGURA 3 - TREINAMENTO DO MODELO DE CLASSIFICAÇÃO



FONTE: Os Autores (2019).

#### 2.1.4 PROCESSAMENTO DE LINGUAGEM NATURAL

De acordo com Beysolow (2018), Processamento de Linguagem Natural ou NLP é um subcampo da ciência da computação focado em compreender linguagem de uma maneira natural, como é feito por humanos, como entender sentimentos contidos em um texto, reconhecimento de fala, entre outros. O NLP se tornou um campo em rápido e constante crescimento, representando boa parte das descobertas em Inteligência Artificial atualmente.

O Processamento de Linguagem Natural possui várias técnicas que podem ser utilizadas de acordo com o objetivo da análise, como *tokenização* de palavras, limpeza de dados de texto, frequência de termos, entre outras. Essas técnicas são

utilizadas na preparação dos conjuntos de dados que serão posteriormente processados pelo algoritmo.

Na área da detecção de notícias falsas, é possível aplicar as técnicas de processamento de linguagem natural em busca de características linguísticas involuntariamente utilizadas por produtores de conteúdo enganoso (MONTEIRO, 2018), fazendo a análise destes dados automaticamente em oposição à análise manual feita por jornalistas e analistas das organizações verificadoras de fatos, como já abordado na Introdução deste documento.

### 2.1.5 BAG OF WORDS

O modelo *Bag-of-Words* é uma maneira de representar dados de texto ao modelar com algoritmos de aprendizagem de máquina. Dados em texto são complexos, impossibilitando sua leitura direta por algoritmos, e por isso o *Bag-of-Words* age simplesmente como um facilitador para o consumo deste tipo de dado por parte do algoritmo, fazendo a contagem de vezes que uma certa palavra aparece em um dado texto (BEYSOLOW, 2018). Esta informação então é inserida em algoritmos e funções de análise, extraíndo características do texto como medida e presença de palavras conhecidas, identificação de padrões em atributos linguísticos, entre outros, para então categorizá-las em dados numéricos que possam ser processados.

Na utilização da *Bag-of-Words* em conjunto com os algoritmos de classificação, as palavras são selecionadas uma a uma a partir do texto e pontuadas de acordo com a característica a ser analisada, e esta pontuação é armazenada em um vetor. Ao aumentar a quantidade de texto a ser analisada, aumenta-se também o tamanho do vetor, podendo gerar um vetor esparso, isto é, com muitas entradas de dados iguais, exigindo mais dos recursos e da memória computacional e podendo apresentar-se como um desafio para alguns algoritmos tradicionais. Para diminuir a quantidade de entradas iguais, podem ser aplicadas algumas técnicas para a limpeza destes dados, ignorando pontuações e palavras frequentes, como preposições e artigos, corrigindo palavras incorretas e reduzindo palavras aos seus radicais, como verbos conjugados. Exemplo: amarão em “amar” ou “ama”, correu em “corre” ou “corr” e assim por diante. Estes tratamentos resultam em um vetor mais conciso e fácil de ser trabalhado e compreendido pelo algoritmo.

Apesar das limitações deste modelo, tais como necessidade de algum tratamento manual do vocabulário, diversidade de resultados e por vezes perda de significado em algumas partes do texto, sua utilização se faz essencial no processamento de dados provenientes de grandes quantidades de texto, que é o caso da Aleteia, que busca informações em um grande número de textos providenciados por inúmeros sites de notícias, trabalhando com as informações tratadas pelo *Bag-of-Words* para processamento e análise dos dados.

#### 2.1.6 WEB CRAWLER E WEB SCRAPING

*Web Crawler* é o nome dado às aplicações de software que vasculham a internet de uma maneira sistematizada, através de tarefas automatizadas (*scripts*) com o objetivo de buscar ou catalogar informações disponíveis na rede mundial de computadores. O *Web Scraping*, por sua vez, consiste em coletar dados através destes *web crawlers*, que executam a requisição de dados ao servidor da página em questão, e então compilam estes dados em um formato para que esta informação seja extraída (MITCHELL, 2015).

Esta prática pode ser considerada um tanto controversa, ao se levar em conta alguns aspectos como direitos autorais do conteúdo e consumo de recursos do servidor da página. Porém o *web crawling* ainda figura em um estágio de “Velho Oeste”, onde o que é e não é permitido ainda está sendo estabelecido (LAWSON, 2015), por isso é sugerido que os *web crawlers* sigam uma série de políticas de boas práticas, como definições de quais páginas podem ser baixadas, frequência com a qual um *web crawler* pode visitar uma página em busca de atualizações, entre outros.

Em um mundo ideal, o *web scraping* não seria necessário, e cada site da web providenciaria sua própria API para compartilhar seus dados de uma maneira estruturada. Porém isso raramente acontece, e quando o site possui uma API para disponibilização de dados, estas são extremamente restritas com relação à quantidade de informação colocada à disposição, e a frequência com a qual podem ser acessados (LAWSON, 2015). Portanto o *web scraping* não apenas circunvê estas limitações, como também oferece uma grande variedade de tecnologias e técnicas de programação para adaptar a ferramenta às necessidades da coleta de dados.

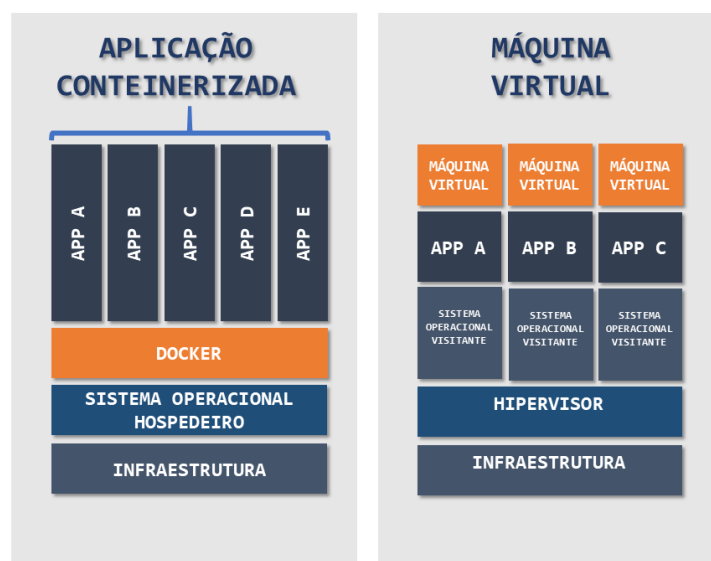
## 2.1.7 CONTEINERIZAÇÃO

Uma boa definição para *Container* (ou Contêiner) seria “um padrão de medida de software”. Um container agrupa todos os códigos, aplicações, ferramentas, bibliotecas e o que mais for necessário para rodar um *software* em uma única unidade, facilitando a movimentação desta unidade toda entre um ambiente de computação para outro, diminuindo drasticamente o tempo de configuração e execução sempre que este *software* for iniciado.

Um Contêiner não é uma Máquina Virtual, de maneira que, para o primeiro, apenas os processos necessários são virtualizados e executados em um ambiente otimizado para este processo específico, sem a necessidade de um *kernel* separado. Já para o segundo, todo o Sistema Operacional é virtualizado, onerando muito mais recursos sem necessariamente utilizá-los (COOK, 2017).

Contêineres e Máquinas Virtuais possuem benefícios similares em isolamento da fonte e alocação, porém no caso dos Contêineres, apenas o que o software precisa para ser executado é o que é virtualizado, e o software sempre será executado uniformemente, independente de estrutura ou ambiente. A figura abaixo conceitua visualmente as diferenças entre uma aplicação containerizada e uma máquina virtual (FIGURA 4):

FIGURA 4 - APLICAÇÃO CONTEINERIZADA E MÁQUINA VIRTUAL



FONTE: Adaptado de DOCKER.COM (2019)

## 2.2 FERRAMENTAS

### 2.2.1 API CHROME

A Google disponibiliza algumas API com diversos propósitos para o desenvolvimento das extensões para o seu navegador Chrome. No caso do desenvolvimento da extensão para o usuário do sistema Aleteia, foram utilizadas as API *chrome.contextMenus* para adicionar o ícone da extensão no menu ao clicar com o botão direito no navegador, na forma de menu de contexto, *chrome.runtime* para limitar a criação do ícone no menu de contexto apenas quando este for solicitado pelo usuário para economia de recursos, e por fim, a *chrome.tabs* para interagir com o sistema da aba em execução, neste caso para executar o script que captura a URL da página e o texto selecionado pelo usuário.

### 2.2.2 TOR

O Projeto Tor é uma ferramenta de roteamento anônimo que encaminha o tráfego de dados através de múltiplos servidores, todos criptografados, protegendo a origem da conexão e dessa forma assegurando privacidade ao usuário.

No início dos anos 90, o projeto começou a ser executado por David Goldschlag, Mike Reed e Paul Syverson no Laboratório de Pesquisas U.S. Naval (NRL), por conta da falta de segurança existente sobre a Internet na época, além da possibilidade de utilização dos dados trafegados para vigilância e rastreamento sem o consentimento dos usuários. Ainda se apoiando no ideal de criação de uma rede descentralizada e diversa, maximizando a transparência e honestidade para com seus usuários, a rede Tor foi disponibilizada em 2002 na forma de licença de código aberto e gratuita. O projeto foi reconhecido pela *Electronic Frontier Foundation* em 2004, a maior organização defensora de privacidade digital e liberdade de expressão conhecida atualmente, e que passou a financiá-lo com o objetivo de dar continuidade ao desenvolvimento do Tor. O projeto figurou como uma ferramenta crucial em acontecimentos históricos pró-liberdade de expressão, como a Primavera Árabe em 2011, nome dado às séries de protestos pró-democracia realizados nos países da Tunísia, Marrocos, Síria, Líbia, Egito e Bahrain, além das revelações de Edward

Snowden em 2013, iniciadas por um debate promovido entre jornalistas sobre vigilância em massa (TOR, 2019).

Através do tipo de conexão assegurado pela rede Tor, é possível efetuar o *web crawling* das páginas respeitando as políticas de revisitação e de etiqueta, que não permitem que a mesma fonte colete os dados de uma certa página múltiplas vezes, ou que acesse muitos dados de uma só vez, utilizando servidores variados para executar estas tarefas.

### 2.2.3 PRIVOXEY E HAPROXY

Os *proxies* Privoxy e Haproxy são ambas soluções de servidores de requisição servidor-cliente para aprimorar o acesso feito pelo *web crawler* através de proteção de alto tráfego e dados e ferramentas de filtragem de informação.

No caso do Privoxy, são aplicados filtros modificadores em páginas *web* e cabeçalhos HTTP, controlando acessos e removendo informações que não serão utilizadas, como propagandas. O Privoxy é um software gratuito em constante aprimoramento e desenvolvimento, possuindo inúmeras opções de configuração, o que o torna altamente flexível para que seja adaptado às necessidades do usuário ou servidor.

Já o Haproxy funciona como balanceador de dados e requisições com o objetivo de não sobrecarregar servidores pequenos ou frágeis, agilizando o consumo e tratamento de dados pelo servidor em aplicações baseadas em HTTP e TCP.

Assim como na Aleteia, outras aplicações necessitam destes modificadores e filtros para que o conteúdo das páginas seja angariado, e estas páginas por sua vez, não bloqueiem o acesso destas aplicações por considerarem muito pesados ou repetitivos.

### 2.2.4 DOCKER

Introduzido em 2013, a plataforma de containerização Docker figura como líder mundial nesta tecnologia, aparecendo em terceiro lugar entre as plataformas de preferência dos desenvolvedores profissionais na pesquisa da base de conhecimento *Stack Overflow* deste ano (STACK OVERFLOW, 2019), atrás apenas dos sistemas operacionais Linux e Windows, sendo essa a primeira vez que tecnologias de

containerização foram inseridas nesta categoria da pesquisa. O Docker oferece ferramentas simples e um pacote universal que engloba todas as dependências de um contêiner, para então ser executado na aplicação *Docker Engine*. Construído em cima do padrão industrial de tecnologias de código aberto, o Docker atende uma grande parte dos profissionais de TI mundialmente, com mais de 1000 imagens de bibliotecas, ambientes de execução, softwares e projetos de código aberto disponíveis para utilização pela comunidade.

A plataforma promete números significativos a nível de negócio, como aumento de mais de 300% na velocidade de disponibilização do software para o mercado, 60% de aumento na velocidade de implantação do software, 40% de redução na infraestrutura de Tecnologia da Comunicação nas empresas, assim como um aumento de 72% na velocidade da resolução de problemas. Atualmente o Docker possui grandes nomes do mercado internacional em sua clientela, como Metlife, PayPal, Visa, GSK, entre outros.

## 2.3 LINGUAGENS DE PROGRAMAÇÃO

### 2.3.1 PYTHON

O Python é uma linguagem de programação orientada a objeto e de alto nível criada em 1991 por Guido Van Rossum com o objetivo de aplicá-la em desenvolvimento *web (server-side)*, desenvolvimento de software, matemática e desenvolvimento de scripts de sistema. Apesar de linguagens de alto nível apresentarem um desempenho inferior às de nível mais baixo ao executar tarefas com grande quantidade de dados, Python continua sendo uma das linguagens mais populares na área de ciência de dados, por seu extenso número de bibliotecas disponíveis e vasta comunidade de desenvolvedores com este mesmo propósito (RASCHKA; MIRJALILI, 2017).

Estão entre algumas das particularidades desta linguagem, a similaridade de sua sintaxe com a língua inglesa pura, além de contar com *indentação* e utilização de espaços em branco para a definição de escopos como funções. O Python também simplifica a finalização de comandos, confiando apenas na criação de uma nova linha com o objetivo de finalizar um comando e iniciar o próximo, em oposição à utilização

do ponto e vírgula para esse mesmo propósito no caso de grande parte das outras linguagens de programação.

Anualmente, o Stack Overflow, a maior comunidade de compartilhamento de conhecimento em desenvolvimento e programação, promove uma pesquisa juntamente aos desenvolvedores que utilizam a sua base de dados, sobre suas preferências com relação às tecnologias de desenvolvimento, bancos de dados, entre outras áreas relacionadas. No questionamento sobre a preferência de linguagens de programação e script, a linguagem Python superou a linguagem Java neste ano, assim como superou as linguagens C# e PHP nos anos anteriores, fazendo dela a linguagem que cresce mais e mais rápido nos últimos anos (STACK OVERFLOW, 2019).

Segundo Kopf (2018), o aumento exponencial na popularidade do Python se dá pela sua facilidade de aprendizado e flexibilidade, sendo utilizada na automação de tarefas simples até importantes análises de grandes quantidades de dados. A linguagem tem se mostrado como um pilar central na Ciência de Dados, uma língua franca na área, com o seu vasto suporte ao Processamento de Linguagem Natural e Aprendizado de Máquina (BEYSOLOW, 2018), ambos utilizados no desenvolvimento da Aleteia.

### 2.3.2 JAVASCRIPT

Javascript é a linguagem de programação da *WEB* utilizada para adicionar comportamentos dinâmicos, armazenar informações e administrar solicitações e respostas em uma página da *web*. É uma linguagem leve, de funções de primeira classe, baseada em protótipo, multi-paradigma, dinâmica, imperativa e declarativa, além de suportar orientação a objetos (FLANAGAN, 2011).

A linguagem JavaScript foi criada com o objetivo de “trazer as páginas da *web* à vida”, por isso no momento de sua criação, ela foi nomeada “LiveScript”. Porém, com o advento da popularidade da linguagem de programação Java na época, seu nome foi alterado para “JavaScript”, no intuito de ser associada à primeira. Mas com a evolução do JavaScript, a linguagem passou a ter suas próprias especificações e características, e hoje em dia já não possui nenhuma ligação com Java.

Os programas utilizados pelo JavaScript são chamados *scripts*, escritos em texto puro, por vezes no próprio HTML da página, executados no momento em que a

página é carregada. Hoje em dia, esta linguagem pode ser executada não apenas no navegador, mas também a nível de servidor ou em qualquer outro dispositivo que possua o programa *JavaScript Engine*. Este programa lê os scripts escritos no código, os compila para a linguagem da máquina, aplicando otimizações ao longo do processo para que o código seja executado com agilidade.

JavaScript é considerada uma linguagem segura por não possuir acesso à memória ou ao processamento do dispositivo, sendo altamente dependente do ambiente no qual o programa está sendo executado. Pelo sétimo ano consecutivo, JavaScript é a linguagem de programação mais popular entre os desenvolvedores profissionais (STACK OVERFLOW, 2019), amplamente utilizada principalmente por oferecer integração total com HTML e CSS, oferecer a possibilidade de ser aplicada tanto na ponta cliente quanto na ponta servidor, suportada pela grande maioria dos navegadores, além de possuir uma grande comunidade de entusiastas pela linguagem que estão sempre em busca de aumentar a base de conhecimento da comunidade e compartilhar novas descobertas, análises, soluções e etc.

## 2.4 BIBLIOTECAS

### 2.4.1 SCIKIT-LEARN

Scikit-Learn é uma biblioteca Python que, segundo Beysolow (2018), todos os cientistas de dados e engenheiros de aprendizado de máquina devem se familiar. Esta biblioteca oferece implementações para algoritmos de aprendizado de máquina e pré-processamento de dados, como o Support Vector Machine, utilizado no desenvolvimento da Aletheia.

A biblioteca é acessada através de uma API e pode ser usada para uma vasta gama de aplicações em tratamento de dados, entre elas Classificação, Regressão, Agrupamento, Redução de Dimensionalidade, Seleção de Modelo e Pré-Processamento. Utilizando suas funções de pré-processamento de dados em conjunto com o modelo *Bag-of-Words*, é possível ajustar os resultados coletados para atender aos objetivos requeridos. O parâmetro padrão da biblioteca conta com a utilização de 5 SVMs, com 1300 a 1400 interações cada.

Por ser uma biblioteca voltada ao aprendizado de máquina, a Scikit-Learn trabalha com algumas estatísticas para compreender melhor a acurácia do modelo

que foi produzido após o treinamento, chamadas *precision*, *recall* e *f1-score*. A estatística *precision* é a relação entre as respostas positivas verdadeiras (pv) contra a soma das positivas verdadeiras (pv) com as falsas verdadeiras (fv), definindo a habilidade do modelo de não produzir falsos negativos. *Recall* é a relação das positivas verdadeiras (pv) com as positivas verdadeiras (pv) somadas com as falsas negativas (fn), informando a habilidade do modelo de classificar todos os elementos positivos. O *F1-score* é a média harmônica entre as estatísticas *precision* e *recall*, mostrando a relação de importância entre elas (TABELA 1):

TABELA 1 - ESTATÍSTICAS SCIKIT-LEARN

<b>Estatísticas</b>	Precision	Recall	F1-score
<b>Fórmulas</b>	$pv / (pv + fv)$	$pv / (pv + fn)$	$F1 = p * f / (p + f)$

FONTE: Os Autores (2019)

#### 2.4.2 NLTK – NATURAL LANGUAGE TOOL-KIT

NLTK é uma das principais bibliotecas Python de Processamento de Linguagem Natural para compreensão de linguagem humana através de *tokenização*, *radicalização*, *pontuação*, *contagem de caracteres*, entre outros.

Se faz necessária sua utilização em conjunto com a biblioteca scikit-Learn, que oferece as funções necessárias para o desenvolvimento de aplicações com características de aprendizado de máquina, mas não provê ferramentas mais específicas para a área de análise de linguagem.

#### 2.4.3 SCRAPY FRAMEWORK

No caso da Aleteia, o *framework* utilizado para efetuar o *web scraping* das notícias foi o *Scrapy*. *Scrapy* é uma biblioteca Python que lida com boa parte da complexidade de encontrar e avaliar links em um site, e de rastreamento de domínios com facilidade (MITCHELL, 2015). Apresenta grande flexibilidade na sua utilização, podendo ser aplicada em arquivamento de históricos, mineração de dados, processamento de informações diversas, entre outros.

A aplicação realiza suas buscas de forma agendada e assíncrona, o que significa que aplicação não precisa aguardar pelos retornos para continuar a

processar as informações, ou então que não irá parar de rodar completamente caso encontre algum erro ou falha, mostrando-se uma ferramenta de *web scraping* muito ágil.

#### 2.4.4 DJANGO PROJECT E DJANGO REST FRAMEWORK

*Frameworks* são estruturas de bibliotecas, componentes e o que for necessário para agrupar em uma só aplicação todas as ferramentas necessárias para o desenvolvimento de um software eliminando esforços repetitivos, erros recorrentes e buscas exaustivas.

Django é um *framework* de código aberto para desenvolvimento *web* em Python, focado em promover um desenvolvimento limpo e rápido através da disponibilização de soluções prontas para as tarefas mais frequentes, como autenticação, administração de conteúdo, *sitemaps*, entre outros. Focado também na segurança, o *framework* auxilia a evitar erros comuns de segurança, como *SQL injection*, falsificação de solicitação entre sites, *clickjacking*, além de prover um modelo seguro de administração de contas de usuários e senhas.

Para o desenvolvimento da Aleteia, foi utilizado o *framework* Django REST, um *framework* focado no desenvolvimento de *APIs web* (Interface de Programação de Aplicação *WEB*), que possui ferramentas mais específicas para este tipo de aplicação, como pacotes de políticas de autenticação, *serializadores* para converter dados complexos como *queries* e instâncias de modelos em dados nativos ao Python que podem ser lidos na forma de *JSON* ou *XML*, possuir uma documentação abrangente e uma comunidade ativa de suporte, além de ser altamente customizável, e utilizado e suportado por grandes nomes da tecnologia mundial como Mozilla, Red Hat, Heroku, entre outros.

#### 2.4.5 JQUERY

Sob o lema “escreva menos, faça mais”, jQuery é a biblioteca de JavaScript mais amplamente utilizada na atualidade, contando com comandos para facilitar a manipulação de documentos HTML, animações, eventos, entre outras aplicações. A biblioteca também possui uma API de fácil utilização para maior entendimento e agilidade para o usuário ao desenvolver com o auxílio de jQuery.

Assim como o JavaScript, o jQuery utiliza o DOM para interagir com o HTML no navegador, desta maneira permitindo que a sintaxe da biblioteca seja utilizada no código JavaScript. Entre as ferramentas incluídas no jQuery estão os Seletores, que indicam sobre quais elementos agir, os Eventos, que definem ocorrências na página, normalmente através de gatilhos, além dos Efeitos, que contém uma ampla gama de efeitos visuais a serem aplicados nos elementos contidos na página.

Com as tecnologias, linguagens, ferramentas e conceitos utilizados no desenvolvimento da Aleteia introduzidos neste capítulo, estabelece-se uma base de conhecimento para que possa ser explicitada a maneira como foram empregadas para a elaboração da aplicação, quais foram os métodos de organização, planejamento e divisão de tarefas, conforme próximo capítulo.

## 3 MATERIAIS E MÉTODOS

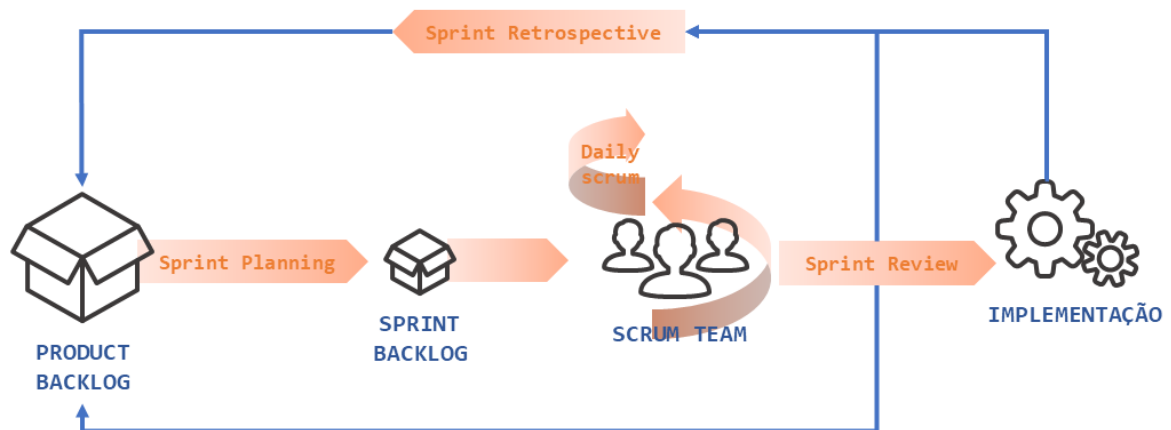
### 3.1 SCRUM

O SCRUM é uma estrutura de trabalho em equipe desenhada para lidar com problemas complexos com eficácia. O método SCRUM aplica o método científico do empirismo juntamente com uma visão holística do processo, e mantém a equipe e o processo sob um planejamento simples e flexível, preparados para lidar com imprevisível.

Segundo Sutherland (2014), um dos criadores do método Scrum, o planejamento é de extrema importância, mas “seguir cegamente os planos é burrice”. O trabalho deve estar aberto a mudanças e inovações, e sempre deve ser revisado, inclusive durante o processo. No método Scrum, se dá menos importância a formulários e procedimentos, e mais importância a criações de valor real que podem ser verificadas em curtos períodos de tempo, e imediatamente aplicadas ao processo.

Para que o método Scrum seja aplicado a um projeto, primeiramente é definido o escopo deste, juntamente com o chamado “*Product Backlog*”, uma lista contendo as funcionalidades do produto que será entregue, isto é, tudo que falta ser desenvolvido no projeto. Com o *product backlog* pronto, é possível planejar as *sprints* através do *Sprint Planning*. *Sprints* são pequenas atividades com o intuito de serem concluídas em curtos períodos de tempo, enquanto o *Sprint Planning*, normalmente realizado em reuniões, é o planejamento das atividades e do tempo dessas *sprints*, e que podem ser alteradas de acordo com o desenvolvimento do projeto. São sugeridas reuniões diárias com a equipe envolvida no projeto, chamadas *Daily scrum*, utilizadas para se discutir o andamento da *sprint*. Após o término de cada *sprint*, é realizada uma *Sprint Review meeting*, uma reunião para apresentar tudo o que foi implementado na *sprint* finalizada, seguida da *Sprint Retrospective*, com o objetivo de planejar a próxima *sprint* (FIGURA 5).

FIGURA 5 - METODOLOGIA SCRUM



FONTE: Adaptado de SCRUM.ORG (2019)

### 3.1.1 CRONOGRAMA E GERENCIAMENTO DE TEMPO

Para o desenvolvimento do sistema, foi adotada a metodologia ágil *SCRUM*, aplicando-se *sprints* com duração de 2 semanas. Esta duração foi fixa durante todo o projeto. Outra medida utilizada pela equipe foram os pontos, que possuíam 3 horas cada, possibilitando que cada atividade a ser desenvolvida dentro dos *sprints* tivessem uma pontuação, aumentando o controle sobre a duração ou nível de dificuldade das tarefas a serem entregadas ao final de cada *sprint*.

Para o gerenciamento dos *sprints* e das tarefas contidas nestes, foram utilizadas duas ferramentas, o sistema de quadro virtual *Trello* e a agenda digital *Google Agenda*. No *Trello* as atividades eram dispostas conforme o seu status de conclusão, assim como demonstravam qual ou quais eram os responsáveis por aquela atividade específica, além de constar a sua pontuação, ou seja, o tempo que levaria para sua execução e conclusão.

As responsabilidades de cada integrante do trabalho foram definidas em “*Back-end*” (lilás), “*Front-end*” (azul) e “*Monografia*” (verde), identificadas visualmente na forma de etiquetas na ferramenta *Trello*, conforme a Figura 6 abaixo:

FIGURA 6 - UTILIZAÇÃO DA FERRAMENTA TRELLO



FONTE: Os Autores (2019)

No Google Agenda, utilizado em compartilhamento por todos da equipe, além da professora orientadora, eram anotadas as datas finais de cada *sprint*, que eram seguidas por uma reunião de fechamento e abertura de *sprints*, além das reuniões com a orientadora do trabalho, conforme a figura 7 abaixo:

FIGURA 7 - UTILIZAÇÃO DO GOOGLE AGENDA

SUN 25	MON 26	TUE 27	WED 28	THU 29	FRI 30	SAT Dec 1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18 ● 7pm Sprint Planning	19	20	21	22 ● 9am Daily Meeting
23	24 Christmas Eve (from 2pm)	25 Christmas Day ● 7pm Daily Meeting	26	27	28	29 ● 9am Daily Meeting ● 5pm Sprint Review
30	31 New Year's Eve (from 2pm) ● 7pm Daily Meeting	Jan 1 New Year's Day ● 7pm Daily Meeting	2 ● 7pm Sprint Planning	3	4	5 ● 9am Daily Meeting

FONTE: Os Autores (2019)

### 3.1.2 SPRINTS

A ideia do projeto Aleteia começou a ser discutida no final de dezembro de 2018, momento no qual foram abertas as contas no Trello e no Google Agenda para acompanhamento de atividades, e no GitHub para compartilhamento de arquivos. Foi enviado também um e-mail à professora orientadora para que fosse possível o agendamento de uma reunião para uma discussão prévia do assunto, o qual foi agendado para fevereiro de 2019, após as férias.

A primeira reunião foi realizada no dia 11 de fevereiro, no qual foram apresentadas as ideias e propostas para o projeto Aleteia, onde foi apresentado um cronograma prévio no próprio Trello, e a partir daí, desenhado o planejamento do desenvolvimento e documentação da aplicação.

Anteriormente à primeira *sprint*, foram realizadas pesquisas sobre o assunto e sobre softwares prévios com este mesmo propósito, além de uma pesquisa técnica intensiva sobre os assuntos de Inteligência Artificial e Aprendizado de Máquina, e teórica na área de *fake News*. Foram disponibilizados cursos através da plataforma Udeemy na área de Redes Neurais para todos os integrantes, o que auxiliou no entendimento da área como um todo e no posterior desenvolvimento do projeto.

TABELA 2 - RESUMO DAS SPRINTS

<b>Sprint</b>	<b>Data</b>	<b>Resumo das Atividades</b>
<b>1</b>	22 de fevereiro	Desenho da arquitetura <i>back-end</i> Desenho da arquitetura <i>front-end</i> Diagrama de Classes Casos de Uso Diagrama de Atividades Refatoração da IA Introdução da Monografia Resumo da Monografia
<b>2</b>	08 de março	Implementação da API Desenvolvimento da Extensão - eventos
<b>3</b>	22 de março	Estudo para atualização da IA
<b>4</b>	05 de abril	Teste da solução de atualização da IA Capítulo 2 da monografia Desenvolvimento da Extensão - visual
<b>5</b>	01 de abril	<i>Dataset</i> para BD Implementação de manutenção da IA Revisão do Capítulo 2 Capítulo 3 da monografia
<b>6</b>	03 de maio	Implementação da estrutura de manutenção
<b>7</b>	17 de maio	Sistema Administrativo Revisão dos Capítulos 2, 3 e 4 da monografia
<b>8</b>	31 de maio	Capítulo 5 da monografia Finalização do treinamento da IA ( <i>webcrawling</i> ) Desenvolvimento da extensão – visual Diagrama de Sequência
<b>9</b>	14 de junho	Revisão geral

FONTE: Os Autores (2019)

### 3.1.2.1 SPRINT 1

A primeira *sprint* foi realizada no dia 22 de fevereiro de 2019, onde, já com a quantidade de pontos definida para a maioria das atividades, foi possível traçar o cronograma de evolução do projeto até a sua entrega. Foram propostos o desenvolvimento das arquiteturas *Back* e *Front-End*, os Diagramas de Atividades e Classes e os Casos de Uso.

A aplicação já estava em processo inicial de desenvolvimento, portanto a proposta era refatorar a SVM para a utilização de Orientação a Objeto, com blocos para treinamento e avaliação.

Para a monografia, a Introdução e o Resumo começaram a ser produzidos, em paralelo com as pesquisas sobre *fake News* e as ferramentas existentes no mercado.

### 3.1.2.2 SPRINT 2

A *sprint* do dia 08 de março focou no desenvolvimento do *front-end* da aplicação, com a definição das funções de leitura de texto, *trigger* de acionamento da extensão no navegador, testes de envio e recebimento de requisição e retorno da mensagem para o usuário.

A parte de *back-end* ficou com a parte de implementação da API entre a aplicação e a extensão, e na monografia foi dada continuação às pesquisas sobre *fake News* e produção de mais conteúdo para a Introdução, além do início da escrita do Capítulo 2.

### 3.1.2.3 SPRINT 3

As tarefas de *front-end* e produção da monografia definidas na *sprint 2* ainda estavam em desenvolvimento pelos participantes responsáveis, enquanto para o *back-end* foi proposto um estudo de possíveis soluções para manter a SVM atualizada com a constante geração de notícias novas.

### 3.1.2.4 SPRINT 4

A *sprint* seguinte ocorreu no dia 5 de abril, e foi utilizada para a realização dos testes da solução proposta para manutenção da atualização da SVM.

Na parte de *front-end*, foi implementado o ícone da extensão no navegador, além do gatilho para acionar a aplicação em si, e um aviso de carregamento como retorno para o usuário.

Na parte da monografia, a professora orientadora iniciou as revisões, e para esta *sprint* já haviam modificações a serem feitas no conteúdo da documentação.

### 3.1.2.5 SPRINT 5

No dia 19 de abril definiu-se que o *dataset* seria movido para um banco de dados e a aplicação deveria ser estruturada para permitir a implementação da manutenção da SVM.

Na monografia, a Fundamentação Teórica recebeu um aprofundamento baseado nos estudos e cursos feitos sobre Inteligência Artificial, Redes Neurais e Aprendizado de Máquina.

#### 3.1.2.6 SPRINT 6

Na *sprint* 6 foi iniciada a implementação da estrutura de base de manutenção da SVM. As partes de *front-end* e *monografia* deram continuidade ao que havia sido iniciado na *sprint* anterior.

#### 3.1.2.7 SPRINT 7

No dia 17 de maio foi definida a organização do sistema administrativo para pré-classificação das notícias no *back-end*, enquanto no *front-end* começaram a ser estudadas as interfaces e identidade visual para o ícone e as notificações.

Na parte de documentação, foi possível avançar para os capítulos de Materiais e Métodos, e também para o capítulo de Resultados.

#### 3.1.2.8 SPRINT 8

A *sprint* 8 foi definida como a *sprint* de início da conclusão do projeto, com a finalização da configuração e estabelecimento do *crawling* na aplicação, treinamento da SVM, adição do botão de acionamento da extensão e início da produção do cronograma geral do projeto.

#### 3.1.2.9 SPRINT 9

Considerada a última *sprint* do projeto, a *sprint* 09 do dia 14 de junho ficou para a realização e documentação dos testes finais da aplicação, e produção dos itens de fechamento da monografia, como os capítulos de Resultados e Conclusão, além da revisão do Resumo.

### 3.2 UML – LINGUAGEM DE MODELAGEM UNIFICADA

A UML ou Linguagem de Modelagem Unificada é uma linguagem visual adotada nacionalmente na área de Engenharia de Software com o objetivo de modelar sistemas computacionais escritos sobre o paradigma de Orientação a Objetos (GUEDES, 2007). Esta modelagem define as características do software, seus requisitos, comportamentos, estrutura lógica e dinâmica, além de definir necessidades infra estruturais para a execução do sistema.

A produção da UML antes do desenvolvimento do software garante uma projeção de Prazos e Custos, e de manutenção também, juntamente com possibilidades de expansão do software, quando este for o caso. Estes documentos definem o escopo e tamanho do projeto, possibilitando a aplicação de possíveis melhorias com relação ao tempo de execução, capital investido, tempo para implantação, entre outros. O Levantamento e Análise de Requisitos, e a Documentação Histórica são de extrema importância ao ofertar um software para um potencial cliente, pois são estes documentos que definirão o futuro deste sistema, como ele deverá ser suportado e qual será a infraestrutura necessária para executá-lo.

Para o sistema Aleteia, o UML foi implementado nos apêndices apresentados neste documento, sendo eles o Diagrama de Casos de Uso (Apêndice C), Diagrama de Classes de Implementação (Apêndice D), Diagramas de Classes (Apêndices E ao L), Diagramas de Atividades (Apêndices M e N), e os Diagramas de Sequência (Apêndice P ao T).

### 3.3 TECNOLOGIAS UTILIZADAS

Inicialmente, a aplicação foi desenvolvida utilizando as ferramentas e tecnologias mais comumente utilizadas no mercado atual, principalmente na área de Aprendizado de Máquina, como a linguagem de programação Python, que já teve sua importância na área de ciência de dados abordada no capítulo anterior, e o *Support Vector Machine*, essencial para análise ao se trabalhar com Aprendizado de Máquina e também já discutido anteriormente. Porém, durante o processo de desenvolvimento, algumas tecnologias foram substituídas por outras mais apropriadas para o caso

específico do projeto, ou então de manutenção mais ágil e facilitada, respeitando o conceito de adaptação do projeto a cada *sprint*, segundo a metodologia Scrum.

A TABELA 3 abaixo demonstra as opções estudadas para cada ferramenta ou tecnologia, e o motivo pelo qual foram selecionadas:

TABELA 3 - OPÇÕES DE TECNOLOGIA PARA O NÚCLEO DA APLICAÇÃO

Processo	Opções	Escolha final	Motivo
<b>Ferramentas para aplicação de SVM com BOW</b>	Pytorch	×	Ferramenta amplamente utilizada com este propósito no mercado atual.
	Tensorflow	×	Também muito utilizada atualmente, possui vasta documentação e comunidade para suporte
	scikit-learn	✓	Utilização mais simples, fácil entendimento, grande economia em linhas de código e flexibilidade nas funções utilizadas no projeto
<b>Ferramentas para <i>web crawling</i></b>	Solução manual por Request (biblioteca Python)	×	Disponibilidade de opções melhores e automatizadas
	Selenium	×	Ferramenta de automatização de testes, que pode ser adaptada para utilização em <i>web crawling</i>
	Scrapy	✓	Padrão de mercado para <i>web crawling</i> , quantidade de código menor, flexibilidade, vasta documentação e comunidade para suporte
<b>Ferramenta de NLP</b>	Spa.CY	×	Preferência do mercado atual na área de NLP, foi utilizada no início do desenvolvimento juntamente com o Post Tags
	NLTK	✓	Possui NLP na língua portuguesa muito mais amadurecido em comparação a outras ferramentas
<b>Extração de características</b>	Post Tags	×	Selecionado inicialmente pela leveza dos arquivos, pois trabalha com vetores na casa de 20 colunas. Porém isso mantém o grau de acurácia em 60% (muito baixo)
	Bag of words	✓	Com vetores na casa de 10 mil colunas, consegue resultados com acurácia de 89%

FONTE: Os Autores (2019)

Na parte de suporte ao desenvolvimento da aplicação, foram utilizadas ferramentas e aplicações com o objetivo de agilizar o processo e torná-lo mais eficiente, escolhidas de acordo com a familiaridade dos desenvolvedores com estas, ou então da compatibilidade com as ferramentas e tecnologias utilizadas no

desenvolvimento. A TABELA 4 abaixo resume estas aplicações de suporte e suas funções no desenvolvimento da aplicação:

TABELA 4 - APLICAÇÕES DE SUPORTE

<b>Aplicação</b>	<b>Função</b>
Mysql	Banco de Dados
Django	Framework de desenvolvimento Python
Django REST Framework	Framework de desenvolvimento Python para API
Jest	Interface para o Sistema Administrativo
Docker	Containerização da aplicação
Tor	Ferramenta de roteamento anônimo para suporte aos <i>web crawlers</i>
Haproxy e Privoxy	Filtros de modificação e balanceamento de dados para os servidores utilizados no <i>web crawling</i>
API Extensão Chrome	APIs disponibilizadas pela Google para o desenvolvimento de extensões para o navegador Google Chrome
jQuery	Biblioteca para a linguagem JavaScript que conta com inúmeros comandos para facilitar a manipulação de documentos HTML, animações, eventos, entre outros.

FONTE: Os Autores (2019)

### 3.3.1 GITHUB

GitHub é uma plataforma de desenvolvimento baseada em ferramentas de controle de versionamento. O controle de versionamento é feito através do acompanhamento das mudanças realizadas no código, software ou arquivo, que podem ser visualizadas por todos os participantes do repositório. Portanto se houver necessidade de retornar a uma versão anterior, esta estará disponível no repositório utilizado, assim como todas as alterações feitas a ela.

Para a utilização do GitHub, primeiramente é feita a criação de um repositório, que abrigará todos os arquivos que estarão sob o controle de versionamento. Neste repositório é possível dar acesso a múltiplos usuários do GitHub, estes que poderão colaborar alterando e atualizando os arquivos. Para cada alteração, é feita uma solicitação, que dependendo do nível de controle pode ser aceita automaticamente ou depender de uma aprovação. Esta solicitação sendo aprovada, o arquivo final é alterado conforme, registrando-se um histórico desta alteração, sendo possível também deixar comentários sobre o que foi feito, conceito de boas práticas utilizado amplamente pelos usuários do GitHub.

Todos os arquivos utilizados no desenvolvimento do Projeto Aleteia foram inseridos em um repositório do GitHub, tanto relativos ao desenvolvimento *front* e

*back-end* da aplicação, quanto aos documentos, imagens e outros conteúdos utilizados na confecção da monografia e documentações. Isso trouxe mais segurança e controle para todos os integrantes do projeto, onde era possível acompanhar independentemente todos as novas edições ou atualizações feitas em quaisquer arquivos do repositório.

Agora com as tecnologias apresentadas, aplicadas juntamente aos métodos comentados neste capítulo, é possível vislumbrar o processo de desenvolvimento da aplicação Aleteia. No próximo capítulo são apresentados os resultados, isto é, a estrutura final da aplicação em si, além de testes e resultados provenientes da finalização do desenvolvimento.

## 4 RESULTADOS

### 4.1 O SISTEMA

Desenhado como um classificador de notícias como verdadeiras ou falsas, através da seleção de um texto pelo usuário, a Aleteia foi desenvolvida na linguagem de programação Python, sendo separada em basicamente três módulos: o módulo de *front-end*, que consiste na extensão do navegador Google Chrome, a qual interage com o usuário, e os dois módulos de *back-end*, o chamado módulo de classificação, que abriga a SVM responsável por classificar as notícias recebidas do módulo de *front-end*, e finalmente o módulo de treinamento, que busca e armazena informações para o treinamento da SVM através de *web crawling*, criando o modelo para classificação.

A extensão foi desenvolvida em JavaScript com o auxílio da biblioteca jQuery e das API da Google, essenciais para a programação de extensões destinadas para utilização no navegador Google Chrome.

Tanto o módulo de classificação de notícias quanto o módulo de treinamento trabalham com o processamento de dados do tipo texto, que são dados com muito ruído, sendo necessário um trabalho de refinamento destes. O framework NLTK de processamento de linguagem natural é responsável pela remoção de palavras de ligação, que não agregam significado ao texto e consumiriam recurso desnecessário de análise caso fossem mantidas, e pela radicalização das palavras, processo que também não influencia no significado final da palavra e ajuda na redução da dimensionalidade do texto como um todo. Esta informação de texto já tratada é trabalhada pela Bag-of-Words, para posterior análise e treinamento pela Support Vector Machine. Estas ferramentas são disponibilizadas pela biblioteca Python Scikit-Learn de implementações para algoritmos de aprendizado de máquina.

O módulo de treinamento contém os modelos, resultados da SVM já treinada com as notícias previamente tratadas conforme processo explicado acima. Estas notícias foram coletadas por *web crawlers* desenvolvidos com o auxílio do framework Scrapy, além do suporte das ferramentas TOR, Privoxy e Haproxy para manter a eficiência e execução destes. Todas estas tecnologias foram previamente abordadas no capítulo 2.

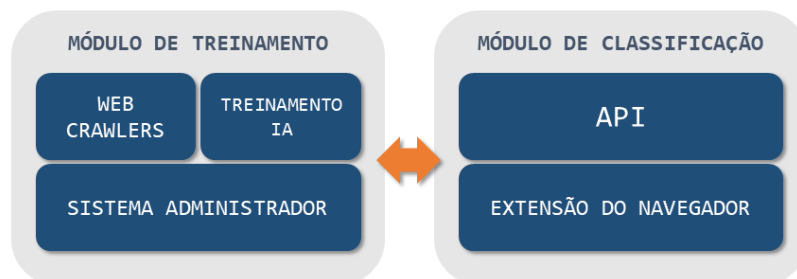
Já o módulo de classificação recebe a notícia enviada pelo usuário pela extensão, através da API, trata o texto conforme o processamento estabelecido e

compara com os módulos treinados previamente pelo módulo de treinamento, retornando a probabilidade para a API.

O framework Django REST assume o papel de API entre os módulos, recebendo e enviando as informações e providenciando a comunicação necessária entre o usuário e o sistema de classificação. Além disso, o framework também concede as ferramentas para a estrutura do Sistema Administrador, juntamente com o *plugin* Jest, que cuida da parte visual, sendo responsável pela interface e design desta parte do sistema.

Com relação ao armazenamento de dados, os modelos, que são os resultados do treinamento do módulo de classificação e que serão cruzados com as informações que o usuário providenciará para a classificação da notícia, são salvos em binários na própria pasta do sistema. Já as notícias coletadas pelo *web crawler* são salvas em um banco de dados MySQL. O Apêndice V desta documentação traz um diagrama de arquitetura demonstrando a estrutura do sistema. A Figura 8 abaixo traz uma simplificação visual da estrutura de módulos da aplicação:

FIGURA 8 - ESTRUTURA DA APLICAÇÃO



FONTE: Os Autores (2019)

## 4.2 O PROCESSO

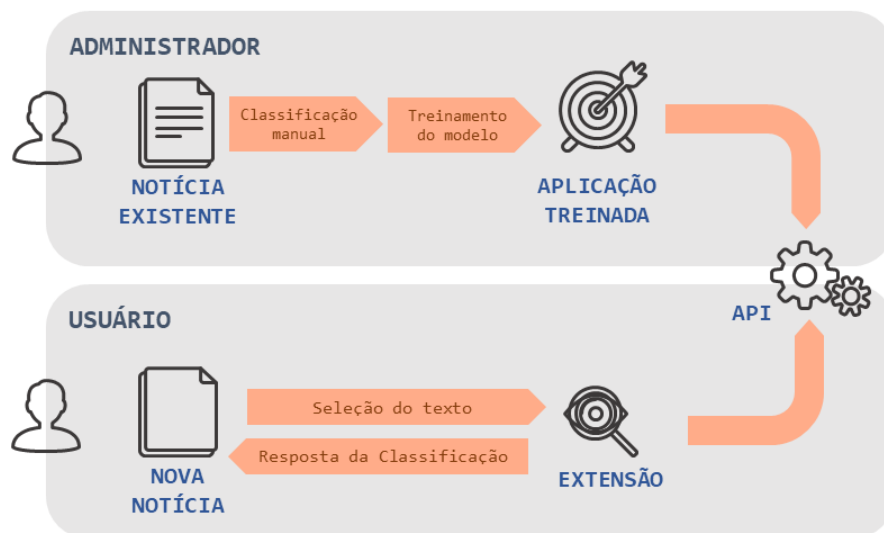
O processo de funcionamento da Aleteia se inicia com a preparação do módulo de treinamento da SVM, que será constante. A angariação de dados para este treinamento se dá através do *web crawler*, fazendo a busca de múltiplas notícias para posterior classificação manual, e então treinamento do módulo de classificação através do SVM em conjunto com o BOW. Esse treinamento produzirá os modelos, que através de comparação serão responsáveis por definir se a notícia selecionada

pelo usuário é verdadeira ou falsa. Os modelos são agrupados por período, filtrando a quantidade e facilitando o serviço de classificação do SVM.

Com a aplicação treinada, o usuário pode utilizar a extensão para selecionar a notícia que deseja verificar a veracidade. Ao selecionar o texto para análise, o sistema informará por meio de um *pop up* que a resposta está sendo carregada, enquanto envia o texto através da API para que a SVM o classifique. A resposta é retornada através do mesmo API e informada ao usuário também na forma de *pop up* quanto à veracidade do texto.

A aplicação também conta com um Sistema Administrador, onde o *web crawler* armazenará as novas notícias coletadas da internet, e uma pessoa será responsável pela validação destas, dentre duas opções que o sistema apresentará, com o objetivo de manter o módulo de classificação sempre treinado e atualizado. Este sistema apresentará os logs das requisições feitas para a API para controle do sistema, além de um painel que mostrará as notícias coletadas pelo *web crawler* e que ainda estão pendentes de classificação manual. Abaixo é apresentado o fluxo de funcionamento da aplicação (FIGURA 9):

FIGURA 9 - FUNCIONAMENTO DA APLICAÇÃO

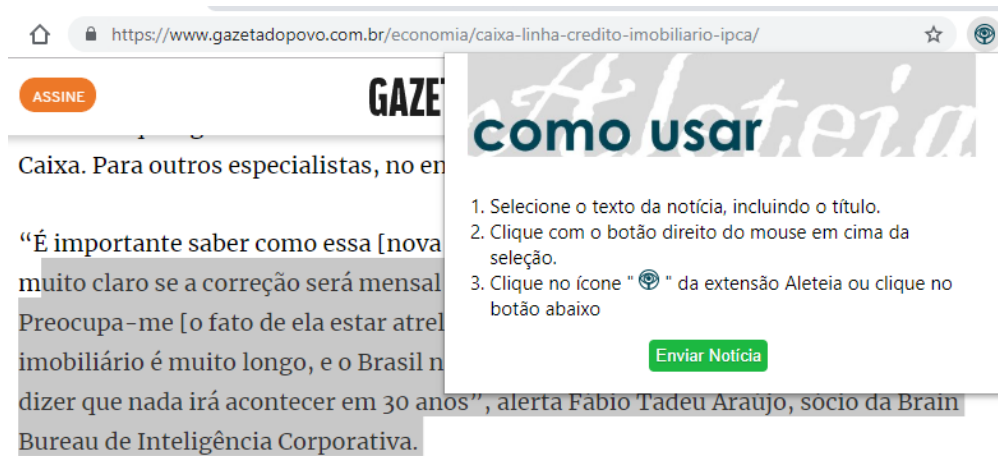


FONTE: Os Autores (2019)

### 4.3 COMO UTILIZAR O SISTEMA

Para a utilização da aplicação Aleteia, o usuário deve ter o navegador Google Chrome e instalar a extensão Aleteia disponível para ele. Ao instalar a extensão, o ícone será mostrado no canto superior direito da tela, contendo as instruções de como utilizá-la (FIGURA 10).

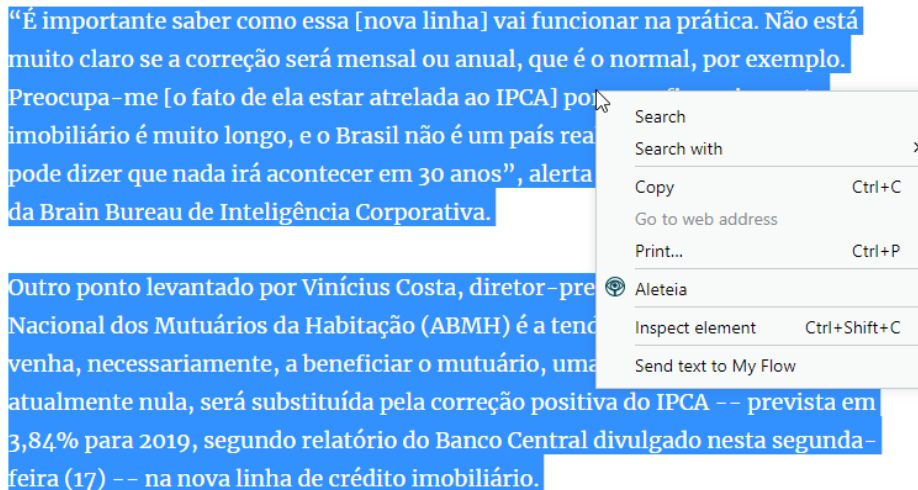
FIGURA 10 - EXTENSÃO: ÍCONE E INSTRUÇÕES



FONTE: Os Autores (2019)

O usuário seleciona com o mouse o texto que deseja verificar, clica com o botão direito sobre o texto selecionado, e dentre as opções figura a opção que envia o texto para verificação (FIGURA 11).

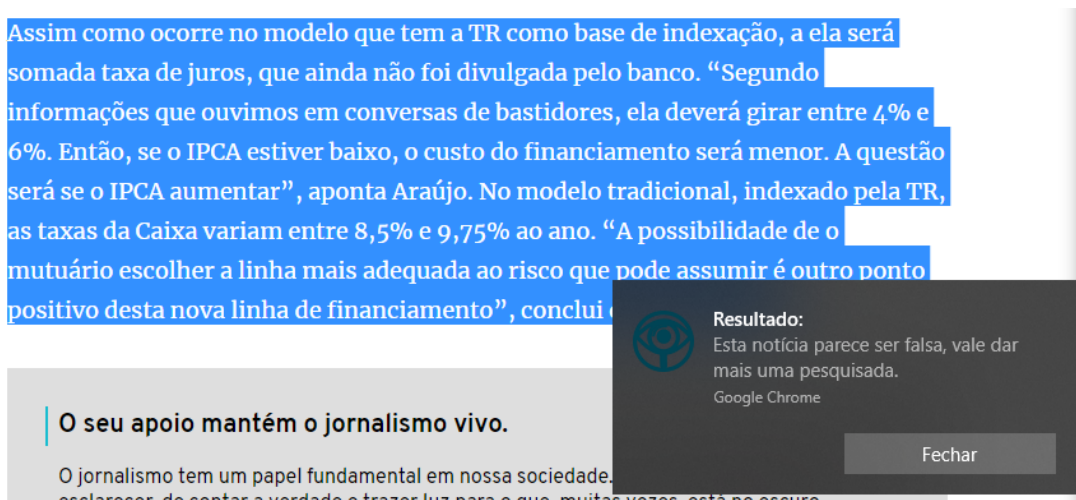
FIGURA 11 - EXTENSÃO: ENVIANDO O TEXTO PARA VERIFICAÇÃO



FONTE: Os Autores (2019)

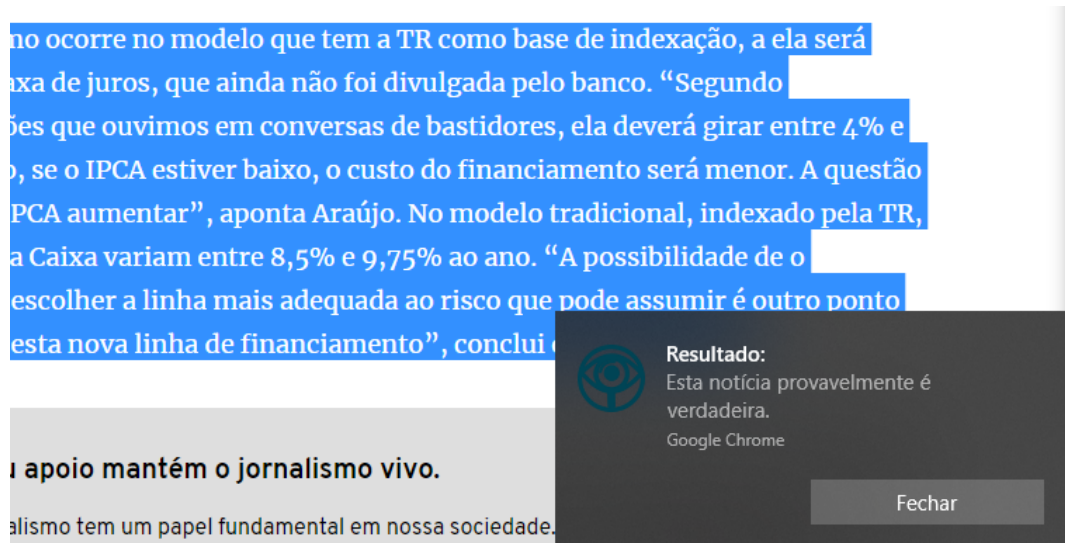
Ao clicar nesta opção, a aplicação avisará através de um *pop up* que a notícia está sendo verificada, e em seguida trará a resposta enviada pelo sistema sobre a veracidade do texto enviado (FIGURAS 12 e 13).

FIGURA 12 - EXTENSÃO: RESULTADO NOTÍCIA FALSA



FONTE: Os Autores (2019)

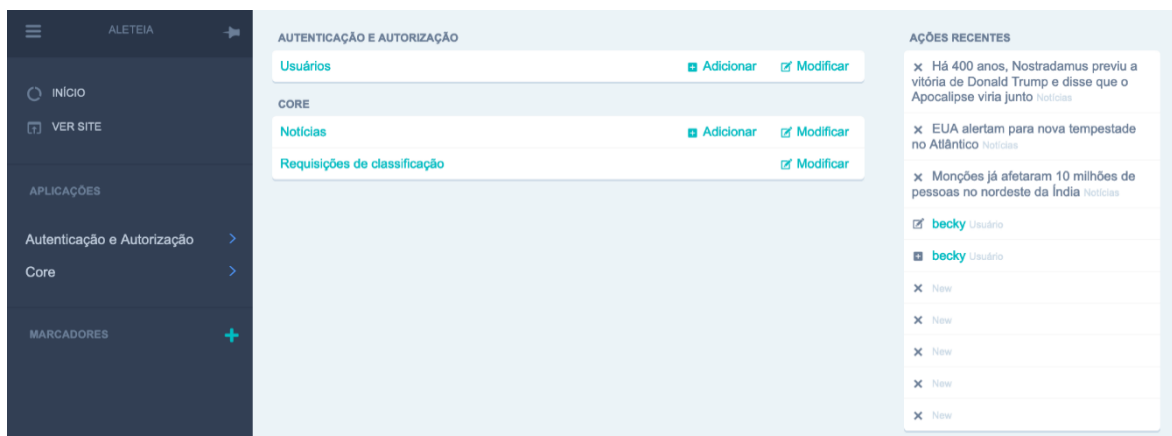
FIGURA 13 - EXTENSÃO: RESULTADO NOTÍCIA VERDADEIRA



FONTE: Os Autores (2019)

No caso do Sistema Administrador, após realizado o login no site, a opção “Core” disponibiliza acesso à dois ambientes: o de Notícias e o de Requisições de Classificação (FIGURA 14).

FIGURA 14 - ADMINISTRADOR: HOME PAGE



FONTE: Os Autores (2019)

O Ambiente de Notícias possui três funções: A primeira consiste na classificação das notícias coletadas pelo *web crawler* manualmente, através de um *dropdown* com três opções para a pergunta “é falsa?”: “desconhecido”, “sim” ou “não”, com o objetivo de atualizar a base de dados do algoritmo de classificação (FIGURA 15).

FIGURA 15 - ADMINISTRADOR: CLASSIFICAÇÃO MANUAL



TITULO	URL	DATA	É FALSA?
Estudo mostra que cada olho focaliza uma letra diferente durante a leitura	https://oglobo.globo.com/sociedade/ciencia/estudo-mostra-que-cada-olho-focaliza-uma-letra-diferente-durante-leitura-4156333	10 de Setembro de 2007 às 00:00	Desconhecido
EUA alertam para nova tempestade no Atlântico	https://oglobo.globo.com/sociedade/ciencia/eua-alertam-para-nova-tempestade-no-atlantico-4156275	10 de Setembro de 2007 às 00:00	Desconhecido
Forte terremoto atinge costa oeste da Colômbia	https://oglobo.globo.com/sociedade/ciencia/forte-terremoto-atinge-costa-oeste-da-colombia-4156263	10 de Setembro de 2007 às 00:00	Desconhecido
Forte terremoto atinge costa oeste da Colômbia	https://oglobo.globo.com/sociedade/ciencia/forte-terremoto-atinge-costa-oeste-da-colombia-4156197	10 de Setembro de 2007 às 00:00	Desconhecido
Forte terremoto atinge a Colômbia	https://oglobo.globo.com/sociedade/ciencia/forte-terremoto-atinge-colombia-4156195	10 de Setembro de 2007 às 00:00	Desconhecido
Corpo de rainha viking morta há 1.200 anos é exumado	https://oglobo.globo.com/sociedade/ciencia/corpo-de-rainha-viking-morta-ha-1200-anos-exumado-4155340	10 de Setembro de 2007 às 00:00	Desconhecido

FONTE: Os Autores (2019)

A segunda opção para adicionar notícias novas manualmente, leva o administrador para uma página onde ele deve preencher as informações da notícia como título, texto, entre outros (FIGURA 16):

FIGURA 16 - ADMINISTRADOR: ADICIONAR NOTÍCIA

The screenshot shows a web form for adding a new news item. The form is titled 'ADICIONAR NOTÍCIAS' and is part of a navigation menu that includes 'INICIO', 'CORE', and 'NOTÍCIAS'. The user 'haruan' is logged in. The form fields are as follows:

- Data:\***: A date selection field with a calendar icon and a time selection icon.
- Título:\***: A text input field.
- Texto:\***: A large text area for the main content of the news item.
- Url:\***: A text input field for the news item's URL.
- É falsa?\***: A dropdown menu currently set to 'Desconhecido'.
- Classificado em:\***: A date selection field with a calendar icon and a time selection icon.

At the bottom of the form, there are three buttons: 'SALVAR' (highlighted in green), 'Salvar e adicionar outro(a)', and 'Salvar e continuar editando'.

FONTE: Os Autores (2019)

E a terceira opção para modificar estes mesmos campos, porém em notícias já existentes no banco de dados da aplicação (FIGURA 17).

FIGURA 17 - ADMINISTRADOR: ALTERAR NOTÍCIAS

INÍCIO > CORE > NOTÍCIAS > FORTE TERREMOTO ATINGE COSTA OESTE DA COLÔMBIA

haruan

< EUA alertam ... Forte terrem... > Histórico

Data:\* 10/09/2007 00:00:00

Título:\* Forte terremoto atinge costa oeste da

Texto:\*  
BOGOTÁ - Um forte terremoto de magnitude 6,8 atingiu a Colômbia na noite de domingo, informou o centro de pesquisas geológicas dos EUA, mas autoridades locais afirmaram que não há relatos de danos graves. Ainda assim, segundo as autoridades de Narino, áreas rurais mais remotas estão sendo contatadas. Moradores disseram à rádio local que o tremor provocou corte de eletricidade em algumas regiões.  
- Por enquanto, as checagens de unidades operacionais não indicam qualquer impacto, mas teremos que esperar pelos resultados do monitoramento - disse Luz Amanda Pulido, diretora nacional da secretaria colombiana de prevenção de desastres.  
Um tremor secundário de magnitude 3,8 sacudiu a área pouco depois do terremoto inicial, de acordo com autoridades. O terremoto ocorreu a uma profundidade de 10 km e seu epicentro foi a 155 km sudoeste da cidade portuária de Buenaventura, segundo o centro dos EUA.

Url:\* Atualmente: <https://oglobo.globo.com/sociedade/ciencia/forte-terremoto-atinge-costa-oeste-da-colombia-4156263>  
Alterar: <https://oglobo.globo.com/sociedade/ciencia/forte-ti>

É falsa? Desconhecido

Classificado em:\*

SALVAR Salvar e adicionar outro(a) Salvar e continuar editando X APAGAR

FONTE: Os Autores (2019)

O ambiente de requisições de classificação mostra todos os textos selecionados para verificação da aplicação, como um log de solicitações (FIGURA 18).

FIGURA 18 - ADMINISTRADOR: REQUISIÇÕES DE CLASSIFICAÇÃO

URL	CRIADO EM	TEXTO	RESPOSTA
<a href="https://www.diariodocentromundo.com.br/sublime-ironia-golpe-implodiu-psdb-e-mdb-e-pt-sobreviveu-com-lula-haddad-per-ricardo-kotscho/">https://www.diariodocentromundo.com.br/sublime-ironia-golpe-implodiu-psdb-e-mdb-e-pt-sobreviveu-com-lula-haddad-per-ricardo-kotscho/</a>	17 de Junho de 2019 às 19:28	Sublime ironia: Golpe implodiu PSDB e MDB, e o PT sobreviveu com Lula/Haddad. Por Ricardo Kotscho Publicado por Diário do Centro do Mundo - 22 de setembro de 2018 ?? ? ? Share ? Lula e Fernando Haddad. Foto: Ricardo Stuckert Publicado originalmente no Bataio do Kotscho POR RICARDO KOTSCHO, jornalista PUBLICIDADE: As redes sociais deram o direito à palavra a uma legião de imbecis que antes falavam apenas em um bar" Umberto Eco, escritor e filósofo italiano, em junho de 2015). *** A 15 dias da eleição, só uma coisa já é certa: PSDB e MDB, os grandes partidos da aliança golpista de 2016, ceivada pela Lava Jato e pela velha mídia, já estão fora do segundo turno, relegados ao bloco dos naticos. E o PT de Lula, o principal alvo da operação para demutar Dilma, sobreviveu com Fernando Haddad, na bica para ir ao segundo turno contra Jair Bolsonaro. A legião de imbecis que ocupou todos os espaços nos últimos anos, ao fazer das redes sociais nativas o campo de combate do antipetismo, não se deu conta de que gerou em seu ventre esta excreção da extrema-direita ululante das viúvas da ditadura. Ficaram pendurados na brocha e só has restou aderir ao capilão e ao general ensandecidos para impedir a quinta vitória consecutiva do PT nas eleições presidenciais. Este é o resumo da ópera bufa lavajatiense, que vai chegando ao seu clímax, depois de Lula comandar da sua cela solitária em Curitiba a derrocada de quem o condenou e agora assiste de camarote à agonia dos seus algozes. Uma cena singular na madrugada de sexta-feira, retratada pela repórter Anna Virginia Baltoussier, na Folha desse sábado, é emblemática desta reta final de campanha. É a foto do tucano Geraldo Alckmin tomando café sozinho, acompanhado apenas de dona Lu, numa lanchonete deserta de beira de estrada, após o debate dos presidenciais na TV Aparecida. Cercado de mesas e cadeiras vazias, sem nenhum militante, assessor, segurança ou misero puxa-saco a seu lado, Alckmin era o símbolo de uma era que acabou. A carta-desespero que FHC enviou aos eleitores na véspera, para tentar ressuscitar a candidatura luciana, pode agora ser colocada na lápide do partido que nos últimos 16 anos se dedicou apenas a destruir o adversário. Para completar o clima de fim de feira da direita golpista, na mesma noite o patético bilionário Henrique Meirelles, candidato só dele mesmo e do que restou do MDB, jogou no ar seu último trunfo: prometeu liberar a maconha. Ainda que não vá para o segundo turno, pois permaneceu aberta a disputa com Ciro Gomes pela segunda vaga, o PT sai desta campanha maior do que entrou, adiando mais uma vez o fim anunciado tantas vezes pela legião de imbecis preconizada por Umberto Eco. Lula sozinho deu um xaque-mate na elite brasileira, no carconido establishment, que entronizou Michel Temer no Palácio do Planalto, e agora junta os cocos de um país dilacerado, quebrado, de volta ao passado de fome, miséria e desemprego. Por onde passa em suas viagens pelo Brasil, seu herdeiro Fernando Haddad é recebido com as mesmas festas que fariam para Lula, se ele pudesse ser candidato, em contraste com seus adversários. Como ele mesmo anunciou na véspera de ser preso, a idela sobreviveu ao homem Lula, condenado sem provas, realimentando a esperança de milhões de brasileiros destituídos de seus direitos básicos de cidadania. Ainda não dá para saber quem vai ganhar, mas já se sabe quem perdeu esta eleição. Entre a volta à ditadura militar de triste memória e o futuro das novas gerações, o país joga o seu destino nas urnas no próximo dia 7 de outubro. Falta pouco agora. Vida que segue.	(response): "Fake"
<a href="https://www.diariodobrasil.org/ptpsdbpmdbforo-de-sp-somente-os-militares-podem-salvar-o-brasil/">https://www.diariodobrasil.org/ptpsdbpmdbforo-de-sp-somente-os-militares-podem-salvar-o-brasil/</a>	17 de Junho de 2019 às 19:28	Jair Bolsonaro e o Exército irão salvar o Brasil do comunismo mais uma vez. A primeira tentativa, em 1930, ficou conhecida pelo nome de INTENTONA COMUNISTA, com a Coluna Prestes. Já a segunda tentativa, em 1964, viria novamente com Prestes conduzindo nos bastidores e João Goulart na presidência. Antes de Jânio Quadros renunciar, ele havia dito que existiam "FORÇAS OCULTAS" trabalhando por trás. Jânio se referia aos comunistas, à ação de CUBA e da URSS (KGB). Antes mesmo do governo FHC, a esquerda já começou a aparelhar o Estado Brasileiro. Com a "redemocratização", isto é, com a entrega do poder aos civis, a esquerda assumiu o controle do país de vez. Desde então, o estado foi sendo cada vez mais aparelhado. O PSDB nunca foi oposição ao PT, sempre foram parceiros. A briga entre eles sempre foi um TEATRINHO e as campanhas do PT dizendo "fora FHC" (e vice-versa) eram apenas conversa pra boi dormir. A elite dos dois partidos sempre souberam disso, pois fazia e ainda faz parte da estratégia gramsciana [...] de ocupação de espaços. Dois partidos fingindo ser inimigos, mas se ajudando debaixo dos panos. Publicamente, eles posam de adversários, e internamente, entre as cúpulas, eles se abraçam e se beijam. A grande massa (que vive alienada em frente as tv's e acredita nas mentiras da grande mídia) e os militares de baixo escollão desses partidos de esquerda ainda são os únicos que acreditam nessa teoria estúpida de que o PSDB seja um inimigo do PT. Com a ajuda da mídia, eles mantiveram essa farsa por décadas, e somente nos últimos 05 ou 06 anos, essa mentira começou a ser desmascarada. O Brasil vive hoje o quase-socialismo, ou um período de pré-socialismo. [...] o país ainda não entrou definitivamente nas trevas, mas está na penumbra. Essa é a terceira tentativa das esquerdas de tentar implantar esse regime no Brasil. PT, PSDB e PMDB aparelharam o Estado e todos os órgãos, instituições e poderes da república. Todos servem apenas ao partido dominante que está no poder e às fações rivais de mesma ideologia que competem com ele.	(response): "Fake"

FONTE: Os Autores (2019)

#### 4.4 TESTES E RESULTADOS OBTIDOS

No Estudo de Detecção de Notícias Falsas para o Português do Programa Institucional de Bolsas de Iniciação Científica, abordado no primeiro capítulo deste documento, foi elaborado um corpus de 7200 notícias datadas de julho a dezembro de 2018, previamente classificadas entre falsas e verdadeiras (em torno de 3500 notícias de cada) (MONTEIRO, 2018). É de grande importância a elaboração de um corpus equilibrado para evitar o sobre ajuste dos dados, tema abordado no capítulo 2 deste documento. Para circunvir esta situação, é preciso que as notícias sejam selecionadas aos pares, sendo ideal que, ao adicionar uma notícia verdadeira sobre um certo tema ao corpus, também seja adicionada uma notícia falsa sobre este mesmo tema. Este corpus escolhido para o treinamento da Aleteia respeita estes princípios.

Os números obtidos com este corpus de notícia, seguindo o conceito do treinamento da biblioteca Scikit-Learn, mostraram os seguintes índices de acurácia:

- Modelo 1: 0.873015873015873
- Modelo 2: 0.8571428571428571
- Modelo 3: 0.8601190476190477
- Modelo 4: 0.873015873015873
- Modelo 5: 0.8908730158730159

O modelo final concluiu os testes com acurácia de 0.8629629629629629, isto é, mais do que 86% de taxa de acertos. Estes resultados são apresentados na forma da chamada Matriz Confusão, conforme abaixo (TABELA 5):

TABELA 5 - MATRIZ CONFUSÃO

	<b>Classificadas pela SVM como Verdadeiras</b>	<b>Classificadas pela SVM como Falsas</b>
<b>Notícias Verdadeiras</b>	850	200
<b>Notícias Falsas</b>	96	1014

FONTE: Os Autores (2019)

A tabela de estatísticas do modelo de treinamento, conforme explanada no capítulo 2, trouxe os seguintes números (TABELA 6):

TABELA 6 - ESTATÍSTICAS DO MODELO

	<b><i>Precision</i></b>	<b><i>Recall</i></b>	<b><i>F1-score</i></b>	<b>Quantidade de entradas</b>
<b>Verdadeira</b>	0.90	0.81	0.85	1050
<b>Falsa</b>	0.84	0.91	0.87	1110

FONTE: Os Autores (2019)

Nos testes realizados com a SVM já treinada, foram analisadas manualmente algumas das notícias nas quais a aplicação classificou erroneamente, onde foi

verificado que, quando o texto possui características opinativas, o grau de subjetividade aumenta, levando a aplicação a tender para a classificação da notícia como falsa. Este é apenas um exemplo que comprova a complexidade do assunto que está sendo abordado, envolvendo inúmeras variáveis ao se tratar de “linguagem humana”, sendo necessário considerá-las detalhadamente ao desenvolver uma ferramenta que precise analisar todos estes elementos.

Com o sistema todo desenhado neste documento, desde os conceitos e teorias estudados para o desenvolvimento, no capítulo 2, passando pelas metodologias utilizadas para aplicar estas ferramentas na execução da aplicação, no capítulo 3, e finalmente resumindo o resultado obtido com o sistema, além de testes comprovando sua eficácia, é possível continuar para as considerações finais deste trabalho, contidas no próximo capítulo.

## 5 CONSIDERAÇÕES FINAIS

A proposta para o desenvolvimento desse sistema nasceu da preocupação com o descaso das pessoas para com a verificação das fontes e fatos das informações compartilhadas por elas mesmas, gerando uma grande onda de desinformação que assola a população atualmente. Através de uma solução baseada sob o conceito de Aprendizado de Máquina e que seria capaz de classificar notícias entre falsas ou verdadeiras por meio de um treinamento prévio apoiado em processamento de linguagem natural, o usuário teria acesso a uma ferramenta que o muniria de informação para que seja quebrado o ciclo do compartilhamento das chamadas *fake news*.

Neste documento foram explicitadas todas as tecnologias, conceitos e ferramentas utilizados para o desenvolvimento desta solução, bem como as metodologias usadas na aplicação destas ferramentas para a execução e desenvolvimento. Estas ferramentas e metodologias fizeram do resultado final possível, delineado no capítulo de Resultados, juntamente com a apresentação das estruturas finais, e os resultados obtidos com os testes aplicados no sistema finalizado.

Por meio da metodologia ágil Scrum e das ferramentas de suporte disponíveis para auxiliar no desenvolvimento da aplicação, foi possível finalizar a aplicação e sua documentação dentro do prazo proposto, apresentando as especificações conforme abordado ao longo do documento.

### 5.1 RECOMENDAÇÃO DE TRABALHOS

Devido às possibilidades de expansão da ferramenta para que esta possa executar a função para a qual foi desenhada mais eficientemente ou ter seu escopo aumentado, ficam aqui propostas algumas melhorias que poderão ser aplicadas em um futuro próximo.

Atualmente, a Aleteia trabalha com apenas um algoritmo principal. A implementação da utilização de múltiplos algoritmos pela aplicação pode aumentar a área de abrangência na qual ela pode ser aplicada, já que atualmente ela cobre

apenas a parte de notícias, podendo estender-se para outros textos como os contidos nas redes sociais, por exemplo.

O Sistema Administrador realiza uma filtragem prévia por período da notícia para facilitar o processo de classificação manual, porém uma filtragem mais assertiva, como por exemplo por assunto, pode reduzir ainda mais o trabalho e tempo gasto neste procedimento.

Outra proposta que refinaria ainda mais os resultados providos pela aplicação seria sua integração a sites verificadores de fatos, até mesmo através do *web crawling*, fazendo com que as notícias utilizadas para treinamento da SVM fossem provenientes de comparações reais e menos subjetivas.

A parte de testes também possui possibilidades de melhoria, como a cobertura de testes unitários na aplicação, e também a implementação de testes *end-to-end* automatizados.

O aumento na quantidade de *web crawlers* trabalhando na busca e angariação de dados para treinamento da aplicação abrangeria ainda mais a área de estudo e desenvolvimento da IA, possibilitando que esta trabalhe mais eficientemente na classificação de diferentes notícias e textos, de diferentes assuntos.

Foi abordada também a abertura de classificação de notícias para o público geral, onde os usuários finais poderiam votar nas notícias pendentes de classificação, auxiliando e agilizando o treinamento da IA, ou também o estudo de possibilidades de automação desta classificação.

E finalmente o desenvolvimento de extensões para outros navegadores, além da adaptação para a utilização como aplicação móvel, integrando redes sociais e aplicativos de mensagem instantânea, possibilitando a leitura e classificação das notícias diretamente a partir destes aplicativos, além de disponibilizar o compartilhamento do resultado através dos mesmos.

## REFERÊNCIAS

AN Introduction to JavaScript. Disponível em <<https://javascript.info/intro>> Acesso em 22 mai. 2019

ARAB Spring. Disponível em <<https://www.history.com/topics/middle-east/arab-spring>>. Acesso em 15 mai. 2019

BEYSOLOW, T., **Applied Natural Language Processing with Python: Implementing Machine Learning and Deep Learning Algorithms for Natural Language Processing**. Estados Unidos: Apress, 2018.

BOWLES, M., **Machine Learning in Python: Essential Techniques for Predictive Analysis**. Estados Unidos: Wiley, 2015.

BROWNLEE, J. **A Gentle Introduction to the Bag-of-Words Model**. Disponível em <<https://machinelearningmastery.com/gentle-introduction-bag-words-model/>>. Acesso em: 22 mai. 2019

CHEQUEADO Automatización: <https://chequeado.com/automatizacion/>. Acesso em 21 mar. 2019.

COOK, J. **Docker for Data Science: Building Scalable and Extensible Data Infrastructure Around the Jupyter Notebook Server**. Estados Unidos: Apress, 2017.

CORTES, C.; VAPNIK, V. **Support-Vector Networks**. Estados Unidos: Kluwer Academic Publishers, 1995.

DA COSTA, M. T. **Fake news tiveram influência na vacinação contra a febre amarela no Brasil, diz chefe da OMS**, 2018 em: <<https://g1.globo.com/bemestar/noticia/fake-news-tiveram-influencia-na-vacinacao-contra-a-febre-amarela-no-brasil-diz-chefe-da-oms.ghtml>>. Acesso em: 21 fev. 2019.

DEVELOPER Survey Results, 2019 Disponível em: <<https://insights.stackoverflow.com/survey/2019>> Acesso em 29 mai. 2019.

DJANGO. Disponível em <<https://www.djangoproject.com/start/overview/>> Acesso em 15 mai. 2019.

DJANGO REST Framework. Disponível em <<https://www.django-rest-framework.org>> Acesso em 15 mai. 2019

DUBBERLEY S., GRIFFIN E., BAL H. M., **Making Secondary Trauma a Primary Issue: A Study of Eyewitness Media and Vicarious Trauma on the Digital Frontline**. Istanbul, 2015.

EVON, D. **Pope Francis Shocks World, Endorses Donald Trump for President**, 2016 em: <<https://www.snopes.com/fact-check/pope-francis-donald-trump-endorsement/>>. Acesso em: 21 fev. 2019.

ÉSOPO, **Fábula 530**.

FACTCHECK.ORG. Disponível em: <<https://en.wikipedia.org/wiki/FactCheck.org>>  
Acesso em: 21 fev 2019.

FLANAGAN, D. **JavaScript: The Definitive Guide**. Estados Unidos: O'reilly, 2011.

FUNKE, D. **There are hundreds of fact-checkers around the world. Here's what some of them look like**, 2018 em: < <https://www.poynter.org/fact-checking/2018/there-are-hundreds-of-fact-checkers-around-the-world-heres-what-some-of-them-look-like/>>

GARBADE, M. J.; **Could Python's Popularity Outperform JavaScript in the Next Five Years?** Disponível em <<https://hackernoon.com/could-pythons-popularity-outperform-javascript-in-the-next-five-years-abed4e307224>> Acesso em 15 mai. 2019

GENERAL Python FAQ. Disponível em  
<<https://docs.python.org/3/faq/general.html#what-is-python>>. Acesso em: 15 mai. 2019

GUEDES, G. T. A. **UML**. Uma Abordagem Prática. Brasil: Novatec, 2007.

HIGA, P. **Facebook tem mais usuários que WhatsApp no Brasil e chega a dois terços da população**: WhatsApp atingiu 120 milhões de usuários brasileiros; Facebook tem 127 milhões, 2018 em <<https://tecnoblog.net/252119/facebook-127-milhoes-usuarios-brasil/>>

HOLAN, A. D. **The Principles of the Truth-O-Meter: PolitiFact's methodology for independent fact-checking**, 2018 em: < <https://www.politifact.com/truth-o-meter/article/2018/feb/12/principles-truth-o-meter-politifacts-methodology-i/#How%20PolitiFact%20Started>>. Acesso em: 23 jan. 2019.

JAVASCRIPT. Disponível em  
<<https://www.codecademy.com/catalog/language/javascript>> Acesso em 22 mai. 2019

JQUERY. Disponível em <<https://jquery.com/>> Acesso em 27 mai. 2019

KOPF, Dan. **The rise of Python, as seen through a decade of Stack Overflow**, 2018 em: <<https://qz.com/1408660/the-rise-of-python-as-seen-through-a-decade-of-stack-overflow/>> Acesso em 01 mai 2019.

LAWSON, R., **Web Scraping with Python**: Scrape data from any website with the power of Python. Birmingham, Mumbai: Packt, 2015.

MACASKILL, E.; DANCE, G. **NSA Files: Decoded**: What the revelations mean for you. Disponível em  
<<https://www.theguardian.com/world/interactive/2013/nov/01/snowden-nsa-files-surveillance-revelations-decoded#section/1>>. Acesso em 04 mai. 2019.

MITCHELL, R., **Web Scraping with Python: Collecting Data from the Modern Web**. Estados Unidos: O'Reilly Media, 2015.

MONTEIRO, R. A., **Detecção Automática de Notícias Falsas para o Português**. São Paulo – USP, 2018. Relatório Final de Projeto.

MONTEIRO, R. A.; SANTOS, R. L. S.; PARDO, T. A. S.; ALMEIDA, T. A.; RUIZ, E. E. S., VALE, O. A. **Contributions to the Study of Fake News in Portuguese: New Corpus and Automatic Detection Results**. São Paulo, 2018.

PYTHON Introduction. Disponível em <[https://www.w3schools.com/python/python\\_intro.asp](https://www.w3schools.com/python/python_intro.asp)>. Acesso em: 15 mai. 2019

RASCHKA, S.; MIRJALILI, V., **Python Machine Learning - Second Edition: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow**. Birmingham, Mumbai: Packt, 2017.

RASCIA, T. **An Introduction to jQuery**. Disponível em <<https://www.digitalocean.com/community/tutorials/an-introduction-to-jquery>> Acesso em 27 mai.

ROBINSON, D. **The Incredible Growth of Python**. Disponível em <<https://stackoverflow.blog/2017/09/06/incredible-growth-python/>>. Acesso em 15 mai. 2019

SCRAPY at a glance. Disponível em <<http://docs.scrapy.org/en/1.6/intro/overview.html>>. Acesso em 22 mai. 2019

SILVA, I. N. et al. **Artificial Neural Networks – A Practical Course**. Brasil: Springer, 2017.

SILVERMAN, C. **This Analysis Shows How Viral Fake Election News Stories Outperformed Real News on Facebook**, 2016 em:

<<https://www.buzzfeednews.com/article/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook>>. Acesso em: 22 mar. 2019.

SUTHERLAND, J. **A Arte de Fazer o Bem do Trabalho na Metade do Tempo**. Estados Unidos: Crown Business, 2014.

TARDÁGUILA, C. **Brasil importou o que houve de pior na eleição dos EUA e no referendo do Brexit**, 2018 em: <<https://piaui.folha.uol.com.br/lupa/2018/09/24/brasil-eua-brexite-noticias-falsas/>>. Acesso em: 21 fev. 2019.

TOR history. Disponível em <<https://www.torproject.org/about/history/>>. Acesso em 22 mai. 2019

WEST, J.; BERGSTROM, C. T.; **Calling Bullshit: Data Reasoning in a Digital World**, 2017 em <<https://callingbullshit.org/>>. Acesso em 22 fev. 2019.

WHAT is Scrum? 2019. Disponível em <<https://www.scrum.org/resources/what-is-scrum>>. Acesso em 22 mai. 2019

ZUBIAGA A., LIAKATA M., PROCTER R., HOI G. W. S., TOLMIE P., **Analysing How People Orient to and Spread Rumours in Social Media by Looking at Conversational Threads**, Inglaterra, 2016.

## APÊNDICE A - HISTÓRIAS DE USUÁRIOS

### **História – Botão em menu de contexto da extensão**

COMO: um usuário da extensão

QUERO: facilidade para enviar a notícia para classificação

PARA: enviar o texto da notícia para classificação

CENÁRIO: O usuário selecionou o texto, com o botão esquerdo do mouse, agora, clicando com o botão direito do mouse no menu do navegador, a opção de envio estará disponível com o ícone da extensão.

### **História – Pop-up explicativo da extensão**

COMO: um usuário da extensão

QUERO: entender o funcionamento da extensão

PARA: conseguir classificar as notícias políticas de sites que acesso com intuito de me informar.

CENÁRIO: estou com a notícia aberta, não sei como utilizar a extensão, cliço no ícone da extensão do navegador para aprender a utilizar a aplicação e obter o resultado da avaliação.

### **História – Ver opções de administrador**

COMO: um usuário administrador

QUERO: ver as ações disponíveis

PARA: decidir qual ação irei executar dentro do sistema

CENÁRIO: Como tenho uma conta de administrador, e possuo privilégios para modificação, exclusão e adição de usuários, classificação de notícias, alteração de classificação e verificação de requisições. Necessito de uma tela que mostre todas as opções disponíveis quando eu conectar ao sistema.

### **História – Gerenciar Administrador (Admin)**

COMO: um usuário administrador

QUERO: incluir usuários e alterá-los

PARA: Adicionar novos administradores e validadores de notícias dentro do sistema, assim, várias pessoas podem ajudar na validação manual das

notícias adicionadas pelo crawler no sistema, permitir que o administrador possa alterar usuários e suas senhas.

CENÁRIO: Um administrador depois de conectado ao sistema, precisa de um botão na sessão de opções de administrador para acessar a página de administrador de usuário,

#### **História – Alterar Notícia (Admin)**

COMO: um usuário administrador

QUERO: alterar a classificação de uma notícia do banco de dados

PARA: Poder alterar informações do texto, URL, data da notícia, entre outras informações

CENÁRIO: Foi cadastrada uma notícia incorreta ou com informações faltantes.

#### **História – Excluir Notícia (Admin)**

COMO: um usuário administrador

QUERO: excluir notícias irrelevantes ou em duplicidade

PARA: excluir notícias que possam estar em duplicidade de alguma forma, ou excluir notícias que foram manualmente inseridas

CENÁRIO: O administrador incluiu erroneamente alguma notícia com muitos itens e textos incorretos, o sistema deve oferecer uma ferramenta dentro da página de busca de notícias, qual o administrador possa excluir uma ou várias notícias ao mesmo tempo

#### **História – Rotular Notícia (Admin)**

COMO: um usuário administrador

QUERO: rotular uma notícia como falsa ou verdadeira

PARA: Com intuito de manter a aplicação sempre atualizada, o usuário administrador deve possuir uma ferramenta qual possa rotular a notícia como falsa ou verdadeira, verificando o contexto, o texto e a fonte de onde ela foi retirada

#### **História – Buscar Notícias (Admin)**

COMO: um usuário administrador

QUERO: buscar dentro do sistema notícias por título, url, ou palavras chaves, ou ainda por classificadas ou não classificadas

PARA: identificar, rotular, alterar ou excluir notícias

CENÁRIO: o administrador necessita classificar notícias para que a I.A se mantenha atualizada, então, ele deve entrar no campo de busca e poder filtrar as notícias do banco pela situação de rótulo, sendo está rotuladas ou não rotuladas

CENÁRIO 2: o administrador está buscando por uma notícia que foi erroneamente classificada como verdadeira, então, ele irá até o filtro de buscas e conseguirá buscar por todas as notícias classificadas como verdadeiras e alterar classificação

### **História – Incluir Notícia (Admin)**

COMO: um usuário administrador

QUERO: incluir notícias manualmente de fontes que estão fora do crawler

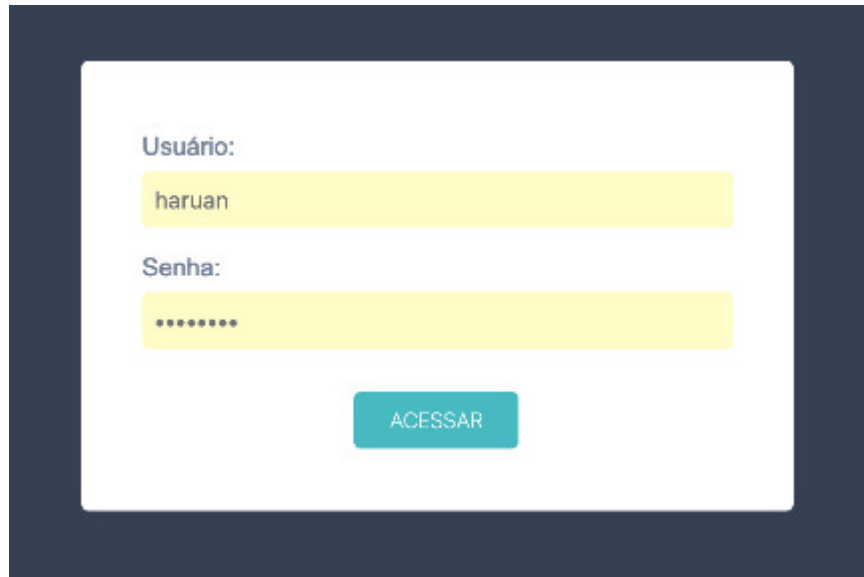
PARA: aumentar a eficácia da classificação da SVM, conseguir adicionar notícias que não são alcançadas pelo crawler e melhorar assim, melhorar o dataset para o aprendizado de máquina.

CENÁRIO: O administrador possui uma notícia que não existe no banco de dados, ele irá até a página de busca de notícias e o sistema proverá uma opção para adicionar a notícia, deverão existir os campos de url fonte, texto, data

## APÊNDICE B - ESPECIFICAÇÕES DE CASOS DE USO

### Especificações de caso de uso – Fazer Login

Data view: DV01 - imagem de login



<b>Nome do UC</b>		<b>Fazer Login</b>
<b>Descrição</b>		Este caso de uso é utilizado para que o administrador possa entrar no sistema de administração de usuários e avaliar notícias
<b>Pré-Condições</b>		Este caso de uso pode iniciar somente se: O visitante estiver conectado à internet.
<b>Ator Primário</b>		Administrador
<b>Fluxo de Eventos Principais</b>		<ol style="list-style-type: none"> <li>1. O usuário acessa o link para login no sistema administrativo.</li> <li>2. O sistema apresenta a DV01.</li> <li>3. O usuário preenche o campo Login e Senha.</li> <li>4. O usuário clica em Login. <b>(E1) (E2)</b></li> <li>5. O sistema valida as informações de login no banco de dados.</li> <li>6. O sistema encaminha o usuário para a tela HOME.</li> <li>7. Fim de caso de uso.</li> </ol>
<b>Fluxos de Exceções</b>	E1. Campo Login e/ou campo senha incorretos.	<ol style="list-style-type: none"> <li>1. O sistema não encontra os dados informados pelo usuário.</li> <li>2. O sistema retorna a mensagem “Por favor, insira um usuário e senha corretos para uma conta de equipe. Note que ambos campos são sensíveis a maiúsculas e minúsculas” ao usuário em vermelho.</li> </ol>

	E2. Campos Login e/ou campo Senha não preenchidos.	<ol style="list-style-type: none"> <li>1. O sistema verifica que o campo de login e/ou o campo senha não está preenchido.</li> <li>2. O sistema retorna uma mensagem</li> </ol>
<b>Regras de Negócio</b>	R1	O login deve ser único no sistema.

### Especificações de caso de uso – Crawler Notícias

<b>Nome do UC</b>	<b>Crawler Notícias</b>	
<b>Descrição</b>	Este caso de uso define como o processo de busca e captura de notícias, para manter a SVM atualizada.	
<b>Pré-Condições</b>	Possuir uma tarefa agendada no servidor de webcrawling	
<b>Ator Primário</b>	Agendador de Tarefas	
<b>Fluxo de Eventos Principais</b>	<ol style="list-style-type: none"> <li>1. O agendador de tarefas executa o comando agendo "python3 -manage.py crawl"</li> <li>2. O sistema consulta o arquivo de configuração para webcrawling.</li> <li>3. O sistema realiza o webcrawling para página de cada site listado dentro do arquivo. (E1)</li> <li>4. O sistema salvará cada página que retornar o resultado desejado no banco de dados.</li> <li>5. Fim de caso de uso</li> </ol>	
<b>Fluxos de Exceções</b>	E1. Acesso a um site retorna um erro	<ol style="list-style-type: none"> <li>1. O sistema retorna uma mensagem de erro ao terminal</li> <li>2. O sistema direciona o processo para o próximo site da lista.</li> </ol>

### Especificações de caso de uso – Treinar Modelo de Classificação

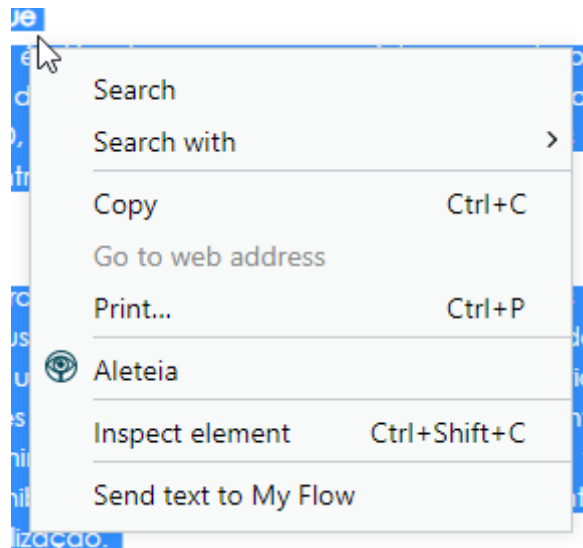
<b>Nome do UC</b>	<b>Treinar Modelo de classificação</b>	
<b>Descrição</b>	Este caso de uso tem como finalidade realizar o treinamento da	
<b>Pré-Condições</b>	Possuir uma tarefa agendada no servidor de webcrawling. Possuir notícias rotuladas por um administrador.	
<b>Ator Primário</b>	Agendador de Tarefas	
<b>Fluxo de Eventos Principais</b>	<ol style="list-style-type: none"> <li>1. O Agendador de tarefas inicia o sistema através do terminal. (R1)</li> </ol>	

	<ol style="list-style-type: none"> <li>2. O sistema consulta as notícias dentro do banco de dados. (R2) (R3)</li> <li>3. O sistema normaliza o texto. (R4)</li> <li>4. O sistema transforma o texto em um array de palavras.</li> <li>5. O sistema realiza a radicalização das palavras da notícia.</li> <li>6. O sistema remove todas as "stopwords" do texto. (R5)</li> <li>7. O sistema utiliza o Bag of Words para vetorizar o array.</li> <li>8. O sistema reduz o tamanho do array para 10350 com as palavras mais importantes.</li> <li>9. O sistema treina a SVM.</li> <li>10. O sistema salva a SVM treinada em 3 arquivos na pasta resources.</li> <li>11. Fim de caso de uso.</li> </ol>										
<b>Regras de Negócio</b>	<table border="1"> <tr> <td data-bbox="550 745 778 840">R1.</td> <td data-bbox="778 745 1455 840">O agendador aciona o sistema todo primeiro dia do mês.</td> </tr> <tr> <td data-bbox="550 840 778 934">R2.</td> <td data-bbox="778 840 1455 934">Somente serão consideradas as notícias dos seis meses anteriores a data de consulta</td> </tr> <tr> <td data-bbox="550 934 778 983">R3.</td> <td data-bbox="778 934 1455 983">As notícias devem possuir rótulo.</td> </tr> <tr> <td data-bbox="550 983 778 1077">R4.</td> <td data-bbox="778 983 1455 1077">Percorre todos os caracteres da notícia transformando-os em caixa baixa.</td> </tr> <tr> <td data-bbox="550 1077 778 2067">R5</td> <td data-bbox="778 1077 1455 2067">Remove todas as seguintes palavras: "de, a, o, que, e, do, da, em, um, para, é, com, não, uma, os, no, se, na, por, mais, as, dos, como, mas, foi, ao, ele, das, tem, à, seu, sua, ou, ser, quando, muito, há, nos, já, está, eu, também, só, pelo, pela, até, isso, ela, entre, era, depois, sem, mesmo, aos, ter, seus, quem, nas, me, esse, eles, estão, você, tinha, foram, essa, num, nem, suas, meu, às, minha, têm, numa, pelos, elas, havia, seja, qual, será, nós, tenho, lhe, deles, essas, esses, pelas, este, fosse, dele, tu, te, vocês, vos, lhes, meus, minhas, teu, tua, teus, tuas, nosso, nossa, nossos, nossas, dela, delas, esta, estes, estas, aquele, aquela, aqueles, aquelas, isto, aquilo, estou, está, estamos, estão, estive, esteve, estivemos, estiveram, estava, estávamos, estavam, estivera, estivéramos, esteja, estejam, estivesse, estivéssemos, estivessem, estiver, estivermos, estiverem, hei, há, havemos, hão, houve, houvermos, houveram, houvera, houveramos, haja, hajamos, hajam, houvesse, houvésssemos, houvessem, houver, houvermos, houverem, houverei, houverá, houveremos, houverão, houveria, houveríamos, houveriam, sou, somos, são,</td> </tr> </table>	R1.	O agendador aciona o sistema todo primeiro dia do mês.	R2.	Somente serão consideradas as notícias dos seis meses anteriores a data de consulta	R3.	As notícias devem possuir rótulo.	R4.	Percorre todos os caracteres da notícia transformando-os em caixa baixa.	R5	Remove todas as seguintes palavras: "de, a, o, que, e, do, da, em, um, para, é, com, não, uma, os, no, se, na, por, mais, as, dos, como, mas, foi, ao, ele, das, tem, à, seu, sua, ou, ser, quando, muito, há, nos, já, está, eu, também, só, pelo, pela, até, isso, ela, entre, era, depois, sem, mesmo, aos, ter, seus, quem, nas, me, esse, eles, estão, você, tinha, foram, essa, num, nem, suas, meu, às, minha, têm, numa, pelos, elas, havia, seja, qual, será, nós, tenho, lhe, deles, essas, esses, pelas, este, fosse, dele, tu, te, vocês, vos, lhes, meus, minhas, teu, tua, teus, tuas, nosso, nossa, nossos, nossas, dela, delas, esta, estes, estas, aquele, aquela, aqueles, aquelas, isto, aquilo, estou, está, estamos, estão, estive, esteve, estivemos, estiveram, estava, estávamos, estavam, estivera, estivéramos, esteja, estejam, estivesse, estivéssemos, estivessem, estiver, estivermos, estiverem, hei, há, havemos, hão, houve, houvermos, houveram, houvera, houveramos, haja, hajamos, hajam, houvesse, houvésssemos, houvessem, houver, houvermos, houverem, houverei, houverá, houveremos, houverão, houveria, houveríamos, houveriam, sou, somos, são,
R1.	O agendador aciona o sistema todo primeiro dia do mês.										
R2.	Somente serão consideradas as notícias dos seis meses anteriores a data de consulta										
R3.	As notícias devem possuir rótulo.										
R4.	Percorre todos os caracteres da notícia transformando-os em caixa baixa.										
R5	Remove todas as seguintes palavras: "de, a, o, que, e, do, da, em, um, para, é, com, não, uma, os, no, se, na, por, mais, as, dos, como, mas, foi, ao, ele, das, tem, à, seu, sua, ou, ser, quando, muito, há, nos, já, está, eu, também, só, pelo, pela, até, isso, ela, entre, era, depois, sem, mesmo, aos, ter, seus, quem, nas, me, esse, eles, estão, você, tinha, foram, essa, num, nem, suas, meu, às, minha, têm, numa, pelos, elas, havia, seja, qual, será, nós, tenho, lhe, deles, essas, esses, pelas, este, fosse, dele, tu, te, vocês, vos, lhes, meus, minhas, teu, tua, teus, tuas, nosso, nossa, nossos, nossas, dela, delas, esta, estes, estas, aquele, aquela, aqueles, aquelas, isto, aquilo, estou, está, estamos, estão, estive, esteve, estivemos, estiveram, estava, estávamos, estavam, estivera, estivéramos, esteja, estejam, estivesse, estivéssemos, estivessem, estiver, estivermos, estiverem, hei, há, havemos, hão, houve, houvermos, houveram, houvera, houveramos, haja, hajamos, hajam, houvesse, houvésssemos, houvessem, houver, houvermos, houverem, houverei, houverá, houveremos, houverão, houveria, houveríamos, houveriam, sou, somos, são,										

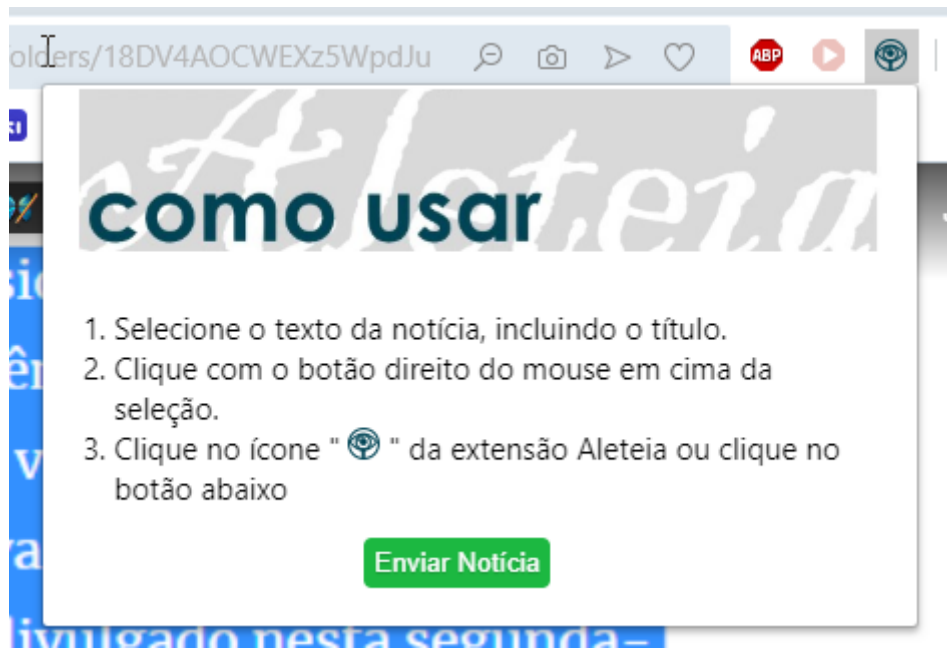
		era, éramos, eram, fui, foi, fomos, foram, fora, fôramos, seja, sejamos, sejam, fosse, fôssemos, fossem, for, formos, forem, serei, será, seremos, serão, seria, seríamos, seriam, tenho, tem, temos, têm, tinha, tínhamos, tinham, tive, teve, tivemos, tiveram, tivera, tivéramos, tenha, tenhamos, tenham, tivesse, tivéssemos, tivessem, tiver, tivermos, tiverem, terei, terá, teremos, terão, teria, teríamos, teriam”
--	--	--

## Especificações de caso de uso – Verificar Veracidade da Notícia

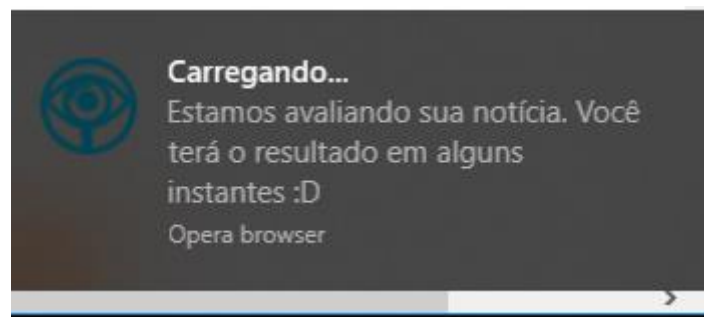
Data views: DV01 – Menu de contexto



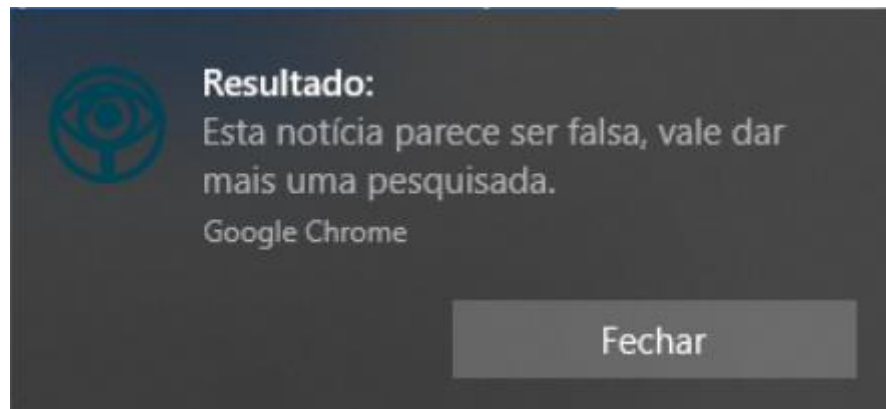
*DV02 – Pop-up da extensão.*



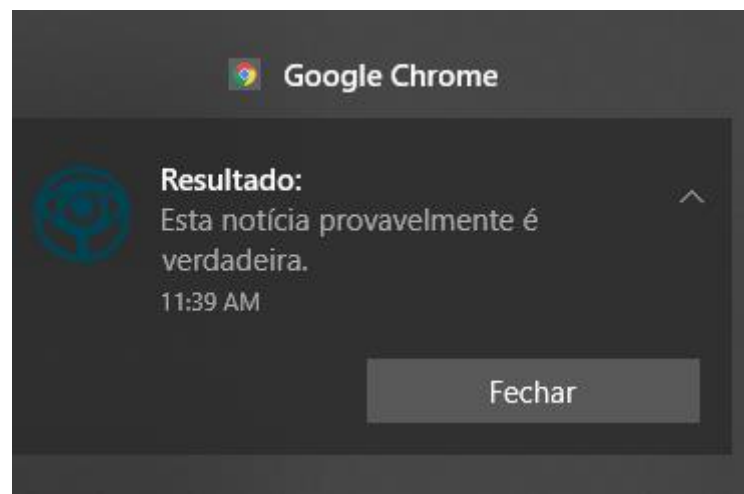
*DV03 – Retorno de Loading.*



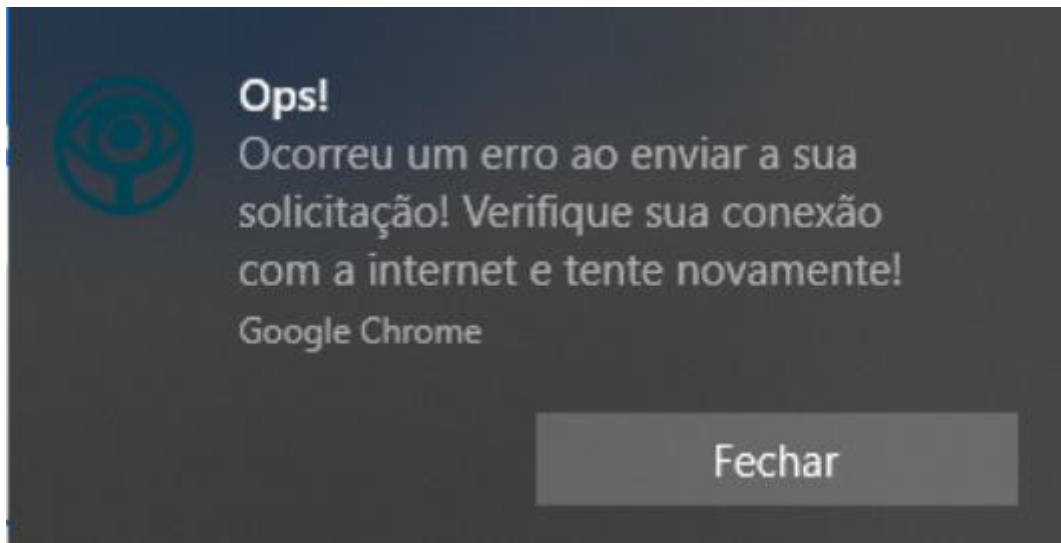
*DV04 – Retorno de Notícia Falsa.*



*DV05 – Retorno de Notícia Verdadeira.*



## DV06 – Erro.



<b>Nome do UC</b>		<b>Verificar Veracidade da Notícia</b>
<b>Descrição</b>		Este caso de uso serve para descrever o processo do usuário interagindo com a extensão para que
<b>Pré-Condições</b>		O usuário precisa estar conectado na internet. O usuário precisa ter a extensão instalada no navegador Google Chrome.
<b>Ator Primário</b>		Usuário
<b>Fluxo de Eventos Principais</b>		<ol style="list-style-type: none"> <li>1. O usuário acessa um site de notícias.</li> <li>2. O usuário seleciona o título e o conteúdo da notícia.</li> <li>3. A extensão habilita o botão no menu de contexto para o usuário. (R1)</li> <li>4. O usuário clica com o botão direito do mouse. (A1)</li> <li>5. O usuário seleciona o botão da Aleteia. (DV01)</li> <li>6. A extensão chama o Caso de Uso "Solicitar Classificação". (E1)</li> <li>7. A extensão envia notificação ao usuário de que sua solicitação foi enviada. (DV03)</li> <li>8. A extensão recebe o retorno do caso de uso.</li> <li>9. A extensão notifica o usuário se a notícia é falsa. (DV04)</li> <li>10. Fim de caso de uso.</li> </ol>
<b>Fluxos Alternativos</b>	A1. Botão da Aplicação no navegador	<ol style="list-style-type: none"> <li>1. O usuário clica no botão com o ícone da extensão na parte superior do navegador.</li> <li>2. O sistema retorna a tela para o usuário. (DV02)</li> <li>3. O usuário clica no botão "Enviar Notícia".</li> </ol>

		4. O sistema retorna ao passo 6 do fluxo principal.
<b>Fluxos de Exceções</b>	E1. Erros no envio na solicitação	1. O sistema não consegue efetuar a solicitação para a API. 2. O sistema retorna uma notificação de erro para o usuário.
<b>Regras de Negócio</b>	R1.	O botão só fica disponível no menu de contexto quando há conteúdo selecionado.

### Especificações de caso de uso – Solicitar Classificação

<b>Nome do UC</b>	<b>Solicitar Classificação</b>
<b>Descrição</b>	Este caso de uso descreve o processo da extensão requisitar a classificação para a API do sistema
<b>Pré-Condições</b>	Um usuário da extensão verificar a veracidade de uma notícia
<b>Ator Primário</b>	Extensão Chrome
<b>Fluxo de Eventos Principais</b>	<ol style="list-style-type: none"> <li>1. A extensão recebe a solicitação do caso de uso Verificar Veracidade da Notícia</li> <li>2. A extensão captura a url da página do usuário e armazena em uma variável.</li> <li>3. A extensão transforma a seleção em string e armazena o texto em uma variável.</li> <li>4. A extensão transforma as variáveis em um objeto JSON.(R1)</li> <li>5. A extensão chama o caso de uso Receber Solicitação.</li> <li>6. A extensão recebe o retorno do caso de uso Receber Solicitação.</li> <li>7. A extensão retorna a informação para o caso de uso Verificar Veracidade da Notícia.</li> <li>8. Fim de caso de uso.</li> </ol>

### Especificações de caso de uso – Classificar Notícia

<b>Nome do UC</b>	<b>Classificar Notícia</b>
<b>Descrição</b>	Este caso de uso explica o processo da recepção da solicitação feita pela extensão para a API
<b>Pré-Condições</b>	Receber uma requisição
<b>Ator Primário</b>	API
<b>Fluxo de Eventos Principais</b>	1. A API recebe os parâmetros enviados pelo caso de uso Solicitar Classificação. (R1)

		<ol style="list-style-type: none"> <li>2. A API executa a classificação da notícia.</li> <li>3. A API envia o resultado por parâmetro para o caso de uso Solicitar Classificação.</li> <li>4. Fim de caso de uso.</li> </ol>
<b>Regras de Negócio</b>	R1	A requisição deve conter os objetos <i>text</i> e <i>url</i> .

## Especificações de caso de uso – Incluir Notícia

Data views: DV01 – Tela home

☰

**AUTENTICAÇÃO E AUTORIZAÇÃO**

Usuários + Adicionar [Modificar](#)

**CORE**

Notícias + Adicionar [Modificar](#)

Requisições de classificação [Modificar](#)

**AÇÕES RECENTES**

+ CAMPO CAMPO Notícias

× ClassificationRequest object (10) Requisição de classificação

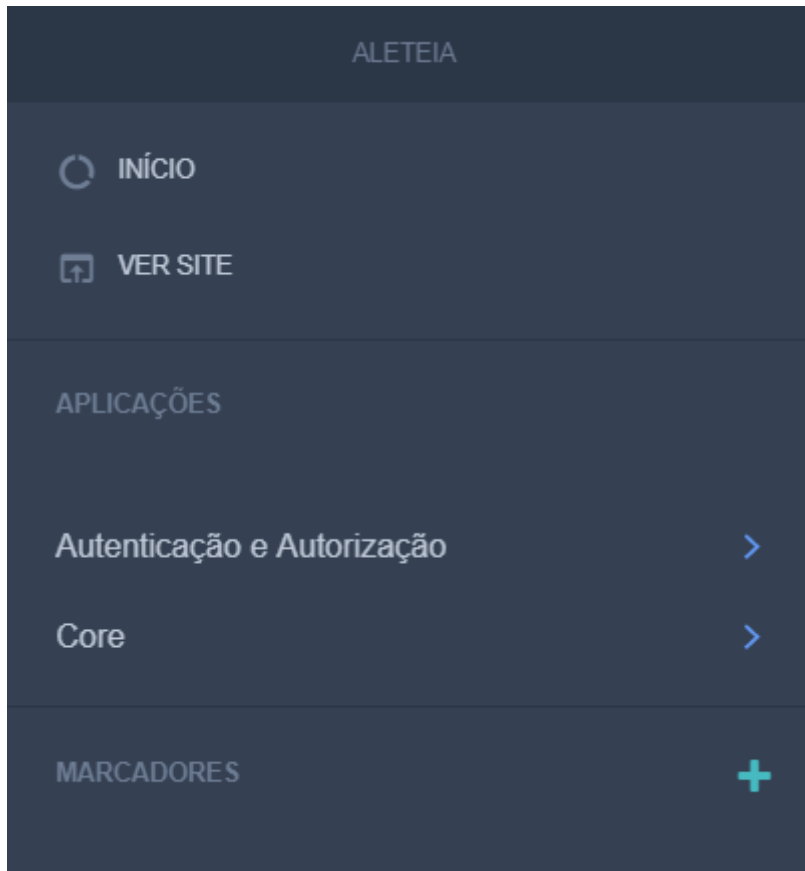
× ClassificationRequest object (11) Requisição de classificação

× ClassificationRequest object (12) Requisição de classificação

× ClassificationRequest object (13) Requisição de classificação



× ClassificationRequest object (14) Requisição de classificação

× ClassificationRequest object (15) Requisição de classificação

*DV02 – Menu Sanduíche*

## DV03 – Tela Adicionar Notícia


INÍCIO > CORE > NOTÍCIAS > ADICIONAR NOTÍCIAS



Data:\*  
   

Título:\*

Texto:\*

Url:\*

Is fake?  
 

Classificado em:\*  
   

**SALVAR**    Salvar e adicionar outro(a)    Salvar e continuar editando

<b>Nome do UC</b>	<b>Incluir notícia</b>
<b>Descrição</b>	Este caso de uso descreve o processo de inclusão manual de uma notícia
<b>Pré-Condições</b>	Usuário administrador estar logado no sistema
<b>Ator Primário</b>	Administrador
<b>Fluxo de Eventos Principais</b>	<ol style="list-style-type: none"> <li>1. O sistema carrega os dados das últimas ações do usuário no sistema.</li> <li>2. O sistema carrega e habilita os botões da sessão Core.</li> <li>3. O sistema carrega e habilita os botões da sessão Autenticação e Autorização.</li> <li>4. O sistema mostra a tela. (DV01)</li> <li>5. O usuário clica no botão adicionar da sessão Core, na linha da Notícia. (A1) (A2) (A3) (A4) (A5) (A6) (A7) (A8).</li> </ol>

		<ol style="list-style-type: none"> <li>6. O sistema carrega e habilita os campos da tela.</li> <li>7. O sistema mostra a tela. (DV03)</li> <li>8. O usuário preenche o campo “Data”. (R1)</li> <li>9. O usuário preenche o campo “Hora”. (R2)</li> <li>10. O usuário preenche os campos “Título”, “Texto” e “URL”.</li> <li>11. O usuário seleciona o rótulo no combo-box “Is Fake?”</li> <li>12. O usuário preenche os campos data e hora do label “Classificado em”. (R1) (R2).</li> <li>13. O usuário clica no botão “Salvar”. (A12) (A13) (E1) (R3)</li> <li>14. O sistema armazena os dados no banco de dados.</li> <li>15. Fim de caso de uso.</li> </ol>
<b>Fluxos Alternativos</b>	A1. Usuário clica no botão “Usuários”	<ol style="list-style-type: none"> <li>1. O sistema chama o caso de uso “Buscar Usuário”.</li> </ol>
	A2. Usuário clica no botão “Modificar” da linha Usuários	<ol style="list-style-type: none"> <li>1. O sistema chama o caso de uso “Buscar Usuário”.</li> </ol>
	A3. Usuário clica no botão “Notícias”	<ol style="list-style-type: none"> <li>1. O sistema chama o caso de uso “Buscar Notícia”.</li> </ol>
	A4. Usuário clica no botão “Requisições de classificação”	<ol style="list-style-type: none"> <li>1. O sistema chama o caso de uso “Buscar Requisições”.</li> </ol>
	A5. Usuário clica no botão “Adicionar” da linha Usuários	<ol style="list-style-type: none"> <li>1. O sistema chama o caso de uso “Adicionar Usuário”</li> </ol>
	A6. Usuário clica em botão “Menu sanduíche”	<ol style="list-style-type: none"> <li>1. O sistema carrega as opções do menu para direcionamento de para as sessões “Core” e “Autenticação e Autorização”.</li> <li>2. O sistema mostra a tela. DV02</li> </ol>
	A7. Usuário clica em “Salvar e adicionar outro(a)”	<ol style="list-style-type: none"> <li>1. O usuário clica em “Salvar e adicionar outro (a)”. (E1)</li> <li>2. O sistema salva os dados no banco de dados.</li> <li>3. O sistema limpa os campos da tela e mostra a tela DV03.</li> <li>4. Retorna ao passo 8 do fluxo principal.</li> </ol>
	A8. Usuário clica em “Salvar e continuar editando”	<ol style="list-style-type: none"> <li>1. O usuário clica em “Salvar e continuar editando”.</li> <li>2. O sistema valida os campos e salva no banco de dados. (E1) (R3)</li> <li>3. O sistema envia uma mensagem de sucesso na parte superior da tela, informando que a notícia foi salva.</li> <li>4. Retorna ao passo 8 do Fluxo principal.</li> </ol>

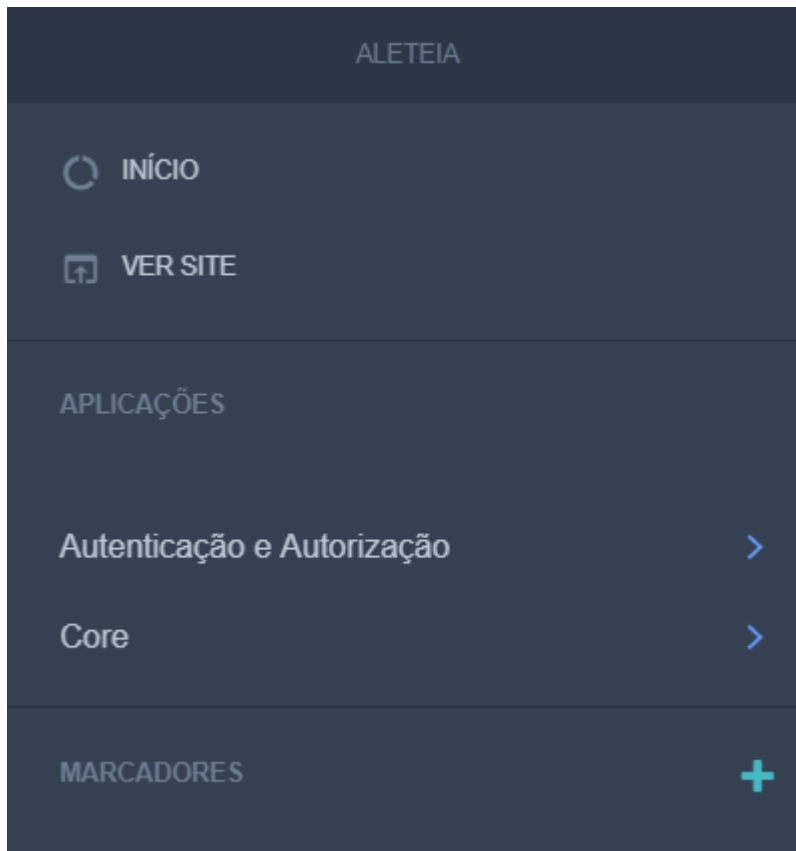
<b>Fluxos de Exceções</b>	E1. Campos sem preenchimento	<ol style="list-style-type: none"> <li>1. O sistema verifica que existem campos sem preenchimento.</li> <li>2. O sistema altera a cor do <i>label</i> de todos os campos sem preenchimento para vermelho.</li> <li>3. O sistema retorna a mensagem “Por favor, corrija os erros abaixo” na tela.</li> <li>4. Retorna ao passo 8 do fluxo principal.</li> </ol>
<b>Regras de Negócio</b>	R1.	O formato requerido é DD/MM/AAAA.
	R2.	O formato requerido é HH:MM.
	R3.	O sistema verifica se todos os campos estão preenchidos e são válidos conforme as regras de negócio.

## Especificações de caso de uso – Buscar Notícia

Data views: DV01 – Tela home

The screenshot displays a web application interface with a dark blue header containing a hamburger menu icon. Below the header, the page is organized into sections:

- AUTENTICAÇÃO E AUTORIZAÇÃO**: A light blue box containing a 'Usuários' entry with '+ Adicionar' and 'Modificar' buttons.
- CORE**: A light blue box containing 'Notícias' (with '+ Adicionar' and 'Modificar' buttons) and 'Requisições de classificação' (with a 'Modificar' button).
- AÇÕES RECENTES**: A white box with a dark blue header containing a list of recent actions:
  - + CAMPO CAMPO Notícias
  - x ClassificationRequest object (10) Requisição de classificação
  - x ClassificationRequest object (11) Requisição de classificação
  - x ClassificationRequest object (12) Requisição de classificação
  - x ClassificationRequest object (13) Requisição de classificação
  - x ClassificationRequest object (14) Requisição de classificação
  - x ClassificationRequest object (15) Requisição de classificação

*DV02 – Menu Sanduíche*

## DV03 – Tela Buscar Notícia

<input type="checkbox"/>	TITULO	URL	DATA	IS FAKE?
<input type="checkbox"/>	Há 400 anos, Nostradamus previu a vitória de Donald Trump e disse que o Apocalipse viria junto	https://www.diariodobrasil.org/ha-400-anos-nostradamus-previu-a-vitoria-de-donald-trump/	3 de Setembro de 201 às 16:54	Desconhecido
<input type="checkbox"/>	Monções já afetaram 10 milhões de pessoas no nordeste da Índia	https://oglobo.globo.com/sociedade/ciencia/moncoes-ja-afetaram-10-milhoes-de-pessoas-no-nordeste-da-india-4156336	10 de Setembro de 2007 às 00:00	Desconhecido
<input type="checkbox"/>	Estudo mostra que cada olho focaliza uma letra diferente durante a leitura	https://oglobo.globo.com/sociedade/ciencia/estudo-mostra-que-cada-olho-focaliza-uma-letra-diferente-durante-leitura-4156333	10 de Setembro de 2007 às 00:00	Desconhecido
<input type="checkbox"/>	Estudo mostra que cada olho focaliza uma letra diferente durante a leitura	https://oglobo.globo.com/sociedade/ciencia/estudo-mostra-que-cada-olho-focaliza-uma-letra-diferente-durante-leitura-4156332	10 de Setembro de 2007 às 00:00	Desconhecido
<input type="checkbox"/>	EUA alertam para nova tempestade no Atlântico	https://oglobo.globo.com/sociedade/ciencia/eua-alertam-para-nova-tempestade-no-atlantico-4156275	10 de Setembro de 2007 às 00:00	Desconhecido
<input type="checkbox"/>	Forte terremoto atinge costa oeste da Colômbia	https://oglobo.globo.com/sociedade/ciencia/forte-terremoto-atinge-costa-oeste-da-colombia-4156263	10 de Setembro de 2007 às 00:00	Desconhecido
<input type="checkbox"/>	EUA alertam para nova tempestade no Atlântico	https://oglobo.globo.com/sociedade/ciencia/eua-alertam-para-nova-tempestade-no-atlantico-4156242	10 de Setembro de 2007 às 00:00	Desconhecido
<input type="checkbox"/>	Forte terremoto atinge costa oeste da Colômbia	https://oglobo.globo.com/sociedade/ciencia/forte-terremoto-atinge-costa-oeste-da-colombia-4156197	10 de Setembro de 2007 às 00:00	Desconhecido
<input type="checkbox"/>	Forte terremoto atinge a Colômbia	https://oglobo.globo.com/sociedade/ciencia/forte-terremoto-atinge-colombia-4156195	10 de Setembro de 2007 às 00:00	Desconhecido

<b>Nome do UC</b>	<b>Buscar notícia</b>
<b>Descrição</b>	Este caso de uso descreve o processo de inclusão manual de uma notícia
<b>Pré-Condições</b>	Usuário administrador estar logado no sistema.
<b>Ator Primário</b>	Administrador
<b>Fluxo de Eventos Principais</b>	<ol style="list-style-type: none"> <li>1. O sistema carrega as notícias do banco de dados.</li> <li>2. O sistema carrega e habilita os botões da tela.</li> <li>3. O sistema mostra a tela. (DV01)</li> <li>4. O usuário clica no nome da Notícia.</li> <li>5. O sistema carrega os</li> <li>6. O usuário clica no botão Notícias. (A1) (A2) (A3) (A4) (A5) (A6) (A7) (A8) (A9)</li> <li>7. O sistema carrega e habilita os campos da tela.</li> <li>8. O sistema carrega a listagem de todas as notícias que existem no banco de dados.</li> <li>9. O sistema mostra a tela. (DV03)</li> </ol>

		<p>10. O usuário clica no combo-box “Por is fake?”. (R1)</p> <p>11. O usuário seleciona a opção desejada.</p> <p>12. O sistema busca no banco de dados pelas notícias que estão rotuladas conforme a seleção do usuário.</p> <p>13. O sistema atualiza a tela com a nova listagem de notícias.</p> <p>14. Fim de caso de uso.</p>
<b>Fluxos Alternativos</b>	A1. Usuário clica no botão “Usuários”	1. O sistema chama o caso de uso “Buscar Usuário”.
	A2. Usuário clica no botão “Modificar” da linha Usuários	1. O sistema chama o caso de uso “Buscar Usuário”.
	A3. Usuário clica no botão “Adicionar” da linha notícias	1. O sistema chama o caso de uso “Buscar Notícia”.
	A4. Usuário clica no botão “Requisições de classificação”	1. O sistema chama o caso de uso “Buscar Requisições”.
	A5. Usuário clica no botão “Adicionar” da linha Usuários	1. O sistema chama o caso de uso “Adicionar Usuário”.
	A6. Usuário clica em botão “Menu sanduíche”	<p>1. O sistema carrega as opções do menu para direcionamento de para as sessões “Core” e “Autenticação e Autorização”.</p> <p>2. O sistema mostra a tela. (DV02)</p>
	A7. Rotular Notícia	<p>1. O usuário clica no combo-box “is fake?”</p> <p>2. O usuário seleciona uma das opções do combo-box para rotular a notícia.</p> <p>3. O usuário repete os passos 1 e 2 quantas vezes desejar.</p> <p>4. O usuário clica no botão “Salvar”.</p> <p>5. O sistema atualiza as notícias no banco de dados.</p> <p>6. O sistema atualiza a tela e retorna a mensagem de sucesso para o usuário.</p>
	A8. Excluir Notícia	<p>1. O usuário seleciona uma ou mais notícias na tela.</p> <p>2. O usuário clica no combo-box na parte inferior da tela. (R2)</p> <p>3. O usuário seleciona a opção “Remover notícias Selecionadas”.</p> <p>4. O usuário clica no botão “Ir”.</p> <p>5. O sistema remove as notícias do banco de dados.</p> <p>6. O sistema atualiza as notícias na tabela e informa o usuário que as notícias foram excluídas.</p>

	A9. Usuário clica no Título de uma notícia.	<ol style="list-style-type: none"> <li>1. O usuário clica no título da notícia.</li> <li>2. O sistema chama o caso de uso Alterar Notícia.</li> </ol>
<b>Regras de Negócio</b>	R1.	<p>O combo-box é carregado com as seguintes opções:</p> <ol style="list-style-type: none"> <li>1. Sim.</li> <li>2. Não.</li> <li>3. Desconhecido.</li> </ol>
	R2.	<p>O combo-box é carregado com as seguintes opções:</p> <ol style="list-style-type: none"> <li>1. Remover Notícias Seleccionadas.</li> <li>2. Classifica Notícias Seleccionadas como verdadeiras</li> <li>3. Classifica Notícias Seleccionadas como falsas</li> </ol>

## Especificações de caso de uso – Alterar Notícia

Data view: DV01 – Alterar Notícia

< Estudo mostr... EUA alertam ... >
Histórico

Data:\*

Título:\*

Texto:\*

Url:\* Atualmente:   
Alterar:

Is fake? Desconhecido

Classificado em:\*

SALVAR
Salvar e adicionar outro(a)
Salvar e continuar editando
✕ APAGAR

<b>Nome do UC</b>		<b>Alterar notícia</b>
<b>Descrição</b>		Este caso de uso serve para descrever o processo de alteração de uma notícia dentro do sistema.
<b>Pré-Condições</b>		Administrador logado no sistema. Administrador na página de Buscar Notícia
<b>Ator Primário</b>		Administrador
<b>Fluxo de Eventos Principais</b>		<ol style="list-style-type: none"> <li>1. O usuário clica na notícia.</li> <li>2. O sistema busca no banco de dados as informações da notícia.</li> <li>3. O sistema carrega a página com os campos preenchidos com as informações.</li> <li>4. O sistema apresenta a tela. (DV01)</li> <li>5. O usuário altera os campos da notícia.</li> <li>6. O usuário clica em "Salvar". (A1) (A2) (A3) (A4)</li> <li>7. O sistema valida as informações da tela. (E1) (R1)</li> <li>8. O sistema salva no banco de dados.</li> <li>9. Fim de caso de uso.</li> </ol>
<b>Fluxos Alternativos</b>	A1. Usuário clica em "Salvar e adicionar outro(a)"	<ol style="list-style-type: none"> <li>1. O usuário clica em "Salvar e adicionar outro (a)". (E1)</li> <li>2. O sistema salva os dados no banco de dados.</li> <li>3. O sistema limpa os campos da tela e mostra a tela DV01.</li> <li>4. Retorna ao passo 4 do fluxo principal.</li> </ol>
	A2. Usuário clica em "Salvar e continuar editando"	<ol style="list-style-type: none"> <li>1. O usuário clica em "Salvar e continuar editando".</li> <li>2. O sistema valida os campos (E1) (R1)</li> <li>3. O sistema envia uma mensagem de sucesso na parte superior da tela, informando que a notícia foi salva.</li> <li>4. Retorna ao passo 4 do Fluxo principal.</li> </ol>
	A3. O usuário clica em "Apagar"	<ol style="list-style-type: none"> <li>1. O usuário clica no botão "Apagar".</li> <li>2. O sistema emite um aviso de confirmação de exclusão para o usuário.</li> <li>3. O usuário clica no botão "Sim, tenho certeza".</li> <li>4. O sistema apaga a notícia do banco de dados.</li> <li>5. O sistema informa o usuário que a notícia foi apagada.</li> <li>6. O sistema chama o caso de uso Buscar Notícia.</li> </ol>
	A4. Histórico	<ol style="list-style-type: none"> <li>1. O usuário clica no botão "Histórico".</li> <li>2. O sistema busca no banco de dados os logs de edição da notícia.</li> <li>3. O sistema retorna na tela as informações de edição da notícia.</li> </ol>
<b>Fluxos de Exceções</b>	E1. Campos incorretos	<ol style="list-style-type: none"> <li>1. O sistema verifica que existem campos sem preenchimento ou preenchidos incorretamente.</li> </ol>

		<ol style="list-style-type: none"> <li>2. O sistema altera a cor do label de todos os campos incorretos, e informa o erro ao usuário.</li> <li>3. O sistema retorna a mensagem “Por favor, corrija os erros abaixo” na tela.</li> <li>4. Retorna ao passo 4 do fluxo principal.</li> </ol>
<b>Regras de Negócio</b>	R1.	O sistema verifica se todos os campos estão preenchidos e são válidos conforme as regras de negócio.

## Especificações de caso de uso – Buscar Usuário

Data view: DV01 – Tela Buscar Usuário

<b>Nome do UC</b>	<b>Buscar Usuário</b>	
<b>Descrição</b>	Este caso de uso descreve o processo de busca de usuários dentro do sistema	
<b>Pré-Condições</b>	O usuário deve estar logado no sistema.	
<b>Ator Primário</b>	Administrador	
<b>Fluxo de Eventos Principais</b>	<ol style="list-style-type: none"> <li>1. O usuário clica no botão “Usuários”.</li> <li>2. O sistema carrega os usuários do banco de dados e preenche a tabela da tela.</li> <li>3. O sistema apresenta a tela. (DV01)</li> <li>4. O usuário preenche o campo “pesquisar”. (A1) (A2) (A3) (A4) (A5)</li> <li>5. O sistema busca pelos usuários no banco de dados.</li> <li>6. O sistema atualiza a tabela de usuários conforme a busca.</li> <li>7. Fim de caso de uso.</li> </ol>	
<b>Fluxos Alternativos</b>	A1. Adicionar Usuário.	<ol style="list-style-type: none"> <li>1. O usuário clica em “Adicionar usuário”.</li> <li>2. O sistema chama o caso de uso Adicionar Usuário.</li> </ol>

	A2. Combo-box “Por Ativo”	<ol style="list-style-type: none"> <li>1. O usuário clica no combo-box.</li> <li>2. O usuário seleciona a opção desejada.</li> <li>3. O sistema busca pelas notícias que estão de acordo com a opção.</li> <li>4. O sistema atualiza a lista de usuários.</li> </ol>
	A3. Combo-box “Por status de super usuário”	<ol style="list-style-type: none"> <li>1. O usuário clica no combo-box.</li> <li>2. O usuário seleciona a opção desejada.</li> <li>3. O sistema busca pelas notícias que estão de acordo com a opção.</li> <li>4. O sistema atualiza a lista de usuários.</li> </ol>
	A4. Apagar Múltiplos usuários	<ol style="list-style-type: none"> <li>1. O usuário seleciona usuários na tela.</li> <li>2. O usuário clica no combo-box localizado na parte inferior da tela.</li> <li>3. O usuário seleciona a opção “Remover usuários selecionados”.</li> <li>4. O usuário clica no botão “Ir”.</li> <li>5. O sistema exclui os usuários do banco de dados.</li> <li>6. O sistema retorna a mensagem de exclusão realizada com sucesso.</li> </ol>

## Especificações de caso de uso – Incluir Usuário

Data view: DV01 – Tela de Inclusão de Usuário

Primeiro, informe um nome de usuário e senha. Depois você será capaz de editar mais opções do usuário.

Usuário:\*

Obrigatório. 150 caracteres ou menos. Letras, números e @/./+/\_ apenas.

Senha:\*

Sua senha não pode ser tão parecida com suas outras informações pessoais.  
 Sua senha precisa conter pelo menos 8 caracteres.  
 Sua senha não pode ser uma senha habitualmente utilizada.  
 Sua senha não pode ser inteiramente numérica.

Confirmação de senha:\*

Informe a mesma senha informada anteriormente, para verificação.

**SALVAR**    Salvar e adicionar outro(a)    Salvar e continuar editando

<b>Nome do UC</b>		<b>Incluir Usuário</b>
<b>Descrição</b>		Este caso de uso descreve a funcionalidade de da inclusão de um usuário no sistema
<b>Pré-Condições</b>		Administrador logado no sistema
<b>Ator Primário</b>		Administrador
<b>Fluxo de Eventos Principais</b>		<ol style="list-style-type: none"> <li>1. O usuário clica em “Adicionar Usuário”</li> <li>2. O sistema carrega e habilita os campos e botões da tela.</li> <li>3. O sistema apresenta a tela. (DV01)</li> <li>4. O usuário preenche os campos usuário, senha e confirmação de senha.</li> <li>5. O usuário clica em salvar. (A1) (A2)</li> <li>6. O sistema valida os campos. (R1) (R2) (R3)</li> <li>7. O sistema salva as informações do usuário do banco de dados.</li> <li>8. O sistema informa que o usuário foi incluído com sucesso.</li> <li>9. Fim de caso de uso.</li> </ol>
<b>Fluxos Alternativos</b>	A1. Usuário clica em “Salvar e adicionar outro(a)”	<ol style="list-style-type: none"> <li>1. O usuário clica em “Salvar e adicionar outro(a)”.</li> <li>2. O sistema executa os passos 6 ao 8 do fluxo principal.</li> <li>3. O sistema retorna ao passo 2 do fluxo principal.</li> </ol>
	A2. Usuário clica em “Salvar e continuar editando”	<ol style="list-style-type: none"> <li>1. O usuário clica em “salvar e continuar editando”.</li> <li>2. O sistema executa os passos 6 ao 8 do fluxo principal.</li> <li>3. O sistema continua na tela de edição do usuário, até o que usuário execute outra ação.</li> </ol>
<b>Regras de Negócio</b>	R1. Campo Usuário	O campo só deve conter letras, números e @/./+/_ e não poderá ultrapassar 150 caracteres
	R2. Campo senha	Este campo deverá mínimo de 8 caracteres, conter algum caractere especial e números.
	R3. Campo confirmação de senha	Este campo deve ser idêntico ao campo senha.

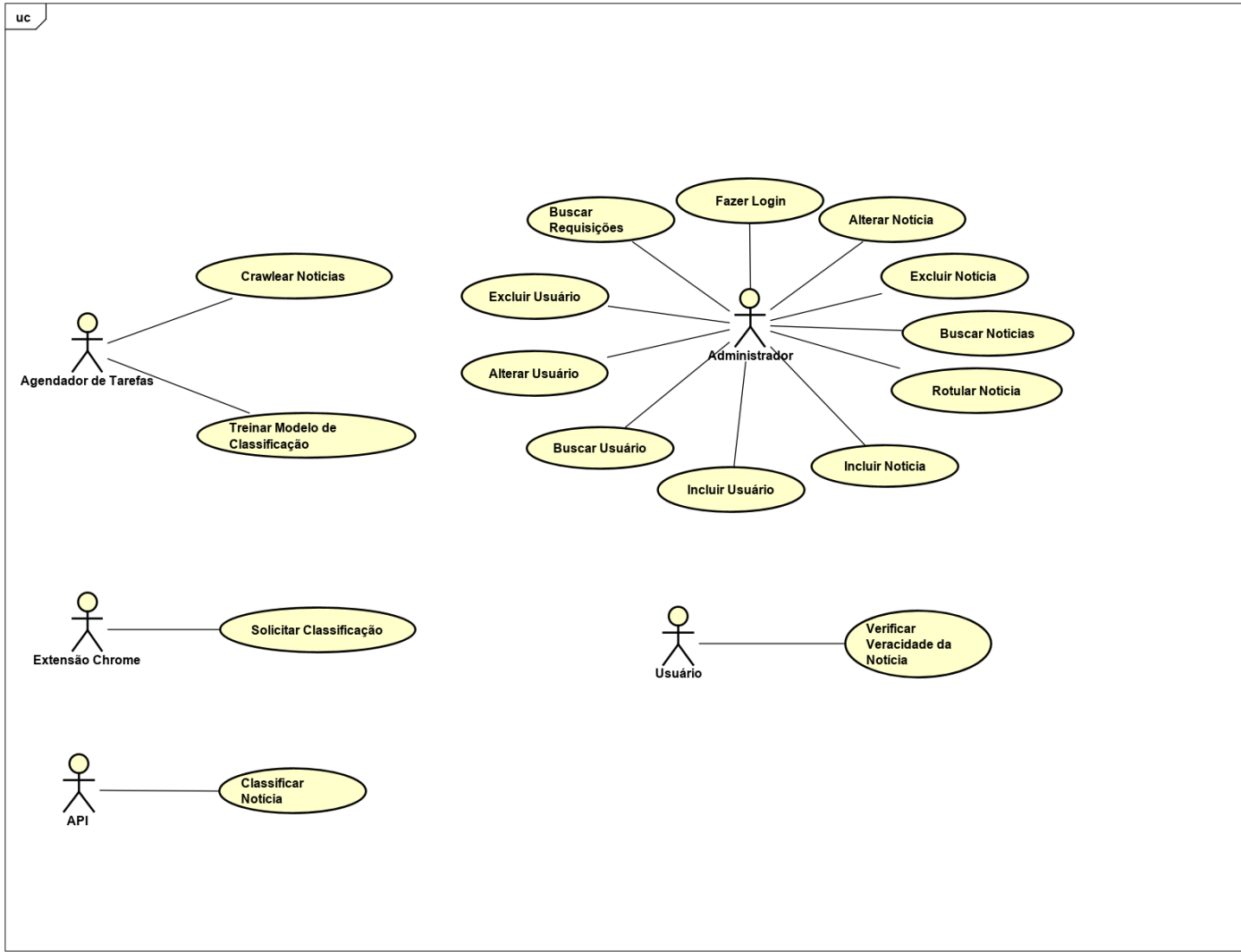
## Especificações de caso de uso – Alterar Usuário

Data view: DV01 – Tela Alterar Usuário

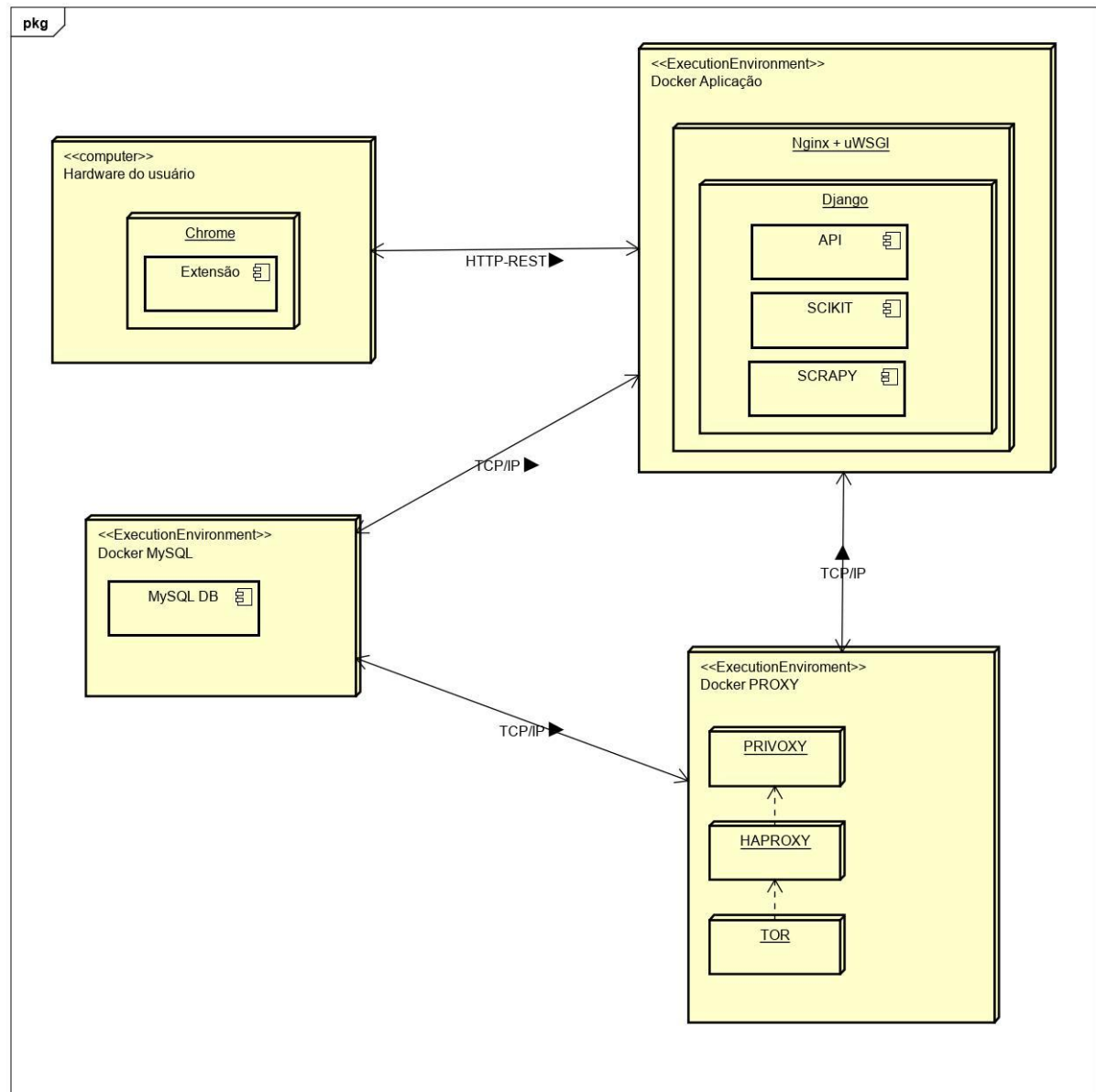
<b>Nome do UC</b>		<b>Alterar Usuário</b>
<b>Descrição</b>		Este caso de uso descreve a funcionalidade de alteração de um usuário no sistema
<b>Pré-Condições</b>		Administrador logado no sistema
<b>Ator Primário</b>		Administrador
<b>Fluxo de Eventos Principais</b>		<ol style="list-style-type: none"> <li>1. O usuário clica em “Modificar Usuário”</li> <li>2. O sistema busca os dados do usuário no banco de dados.</li> <li>3. O sistema preenche a tela com os dados do usuário.</li> <li>4. O sistema apresenta a tela. (DV01)</li> <li>5. O usuário altera os campos desejados.</li> <li>6. O usuário clica em salvar. (A1) (A2)</li> <li>7. O sistema valida os campos. (R1) (R2) (R3)</li> <li>8. O sistema salva as informações do usuário do banco de dados.</li> <li>9. O sistema informa que o usuário foi alterado com sucesso.</li> <li>10. Fim de caso de uso.</li> </ol>
<b>Fluxos Alternativos</b>	A1. Usuário clica em “Salvar e adicionar outro(a)”	<ol style="list-style-type: none"> <li>4. O usuário clica em “Salvar e adicionar outro(a)”.</li> <li>5. O sistema executa os passos 6 ao 8 do fluxo principal.</li> <li>6. O sistema retorna ao passo 2 do fluxo principal.</li> </ol>
	A2. Usuário clica em “Salvar e continuar editando”	<ol style="list-style-type: none"> <li>4. O usuário clica em “salvar e continuar editando”.</li> <li>5. O sistema executa os passos 6 ao 8 do fluxo principal.</li> <li>6. O sistema continua na tela de edição do usuário, até o que usuário execute outra ação.</li> </ol>
	A3. Excluir Usuário	<ol style="list-style-type: none"> <li>1. O usuário clica no botão “Apagar”.</li> <li>2. O sistema retorna uma confirmação de exclusão para o usuário.</li> </ol>

		<ol style="list-style-type: none"> <li>3. O usuário clica em “Confirmar Exclusão de usuário”.</li> <li>4. O sistema apaga o usuário do banco de dados.</li> <li>5. O sistema chama o caso de uso Buscar Usuário.</li> </ol>
<b>Regras de Negócio</b>	R1. Campo Usuário	O campo só deve conter letras, números e @/./+/_ e não poderá ultrapassar 150 caracteres
	R2. Campo senha	Este campo deverá mínimo de 8 caracteres, conter algum caractere especial e números.
	R3. Campo confirmação de senha	Este campo deve ser idêntico ao campo senha.

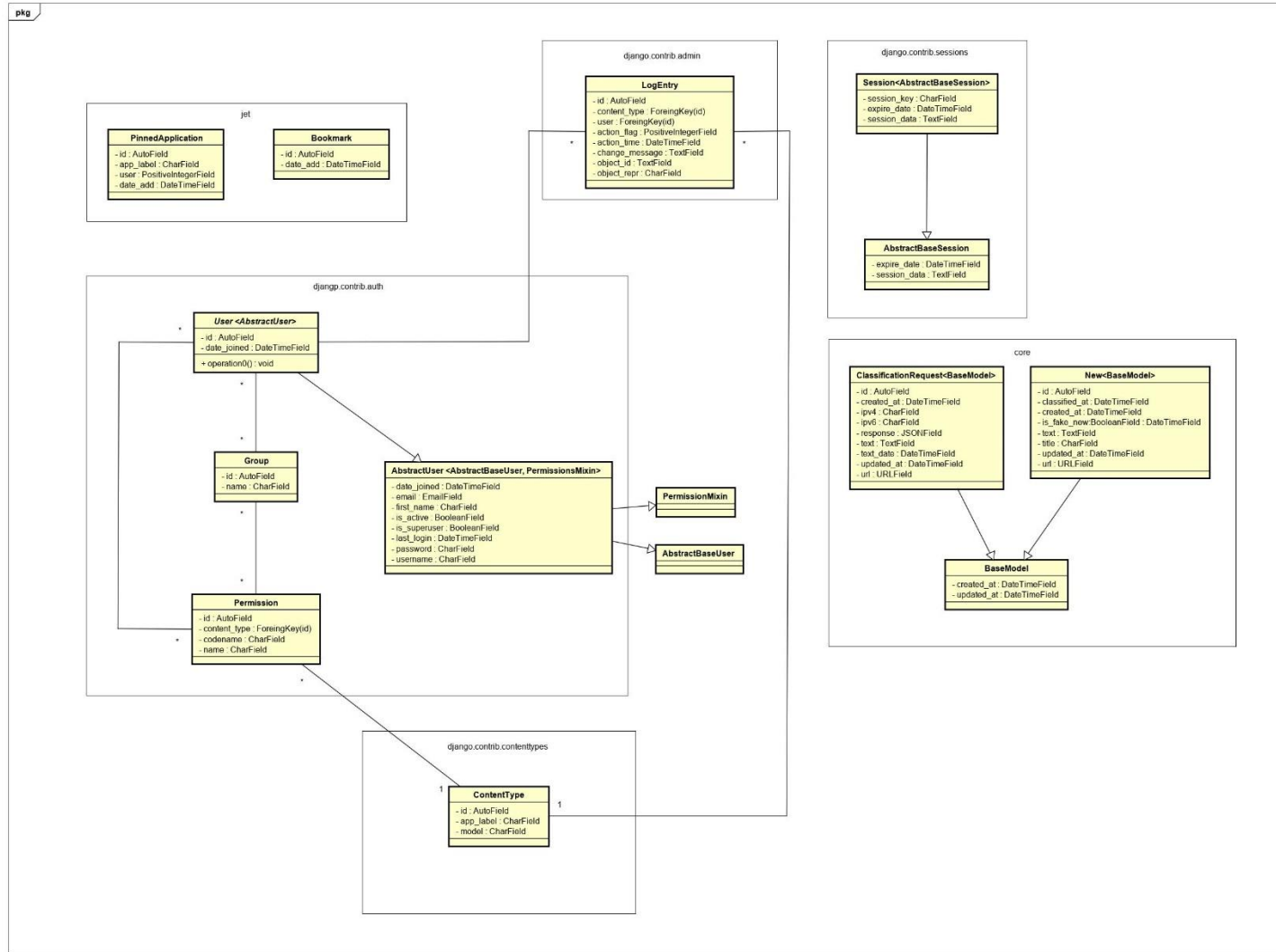
### APÊNDICE C - DIAGRAMA DE CASOS DE USO



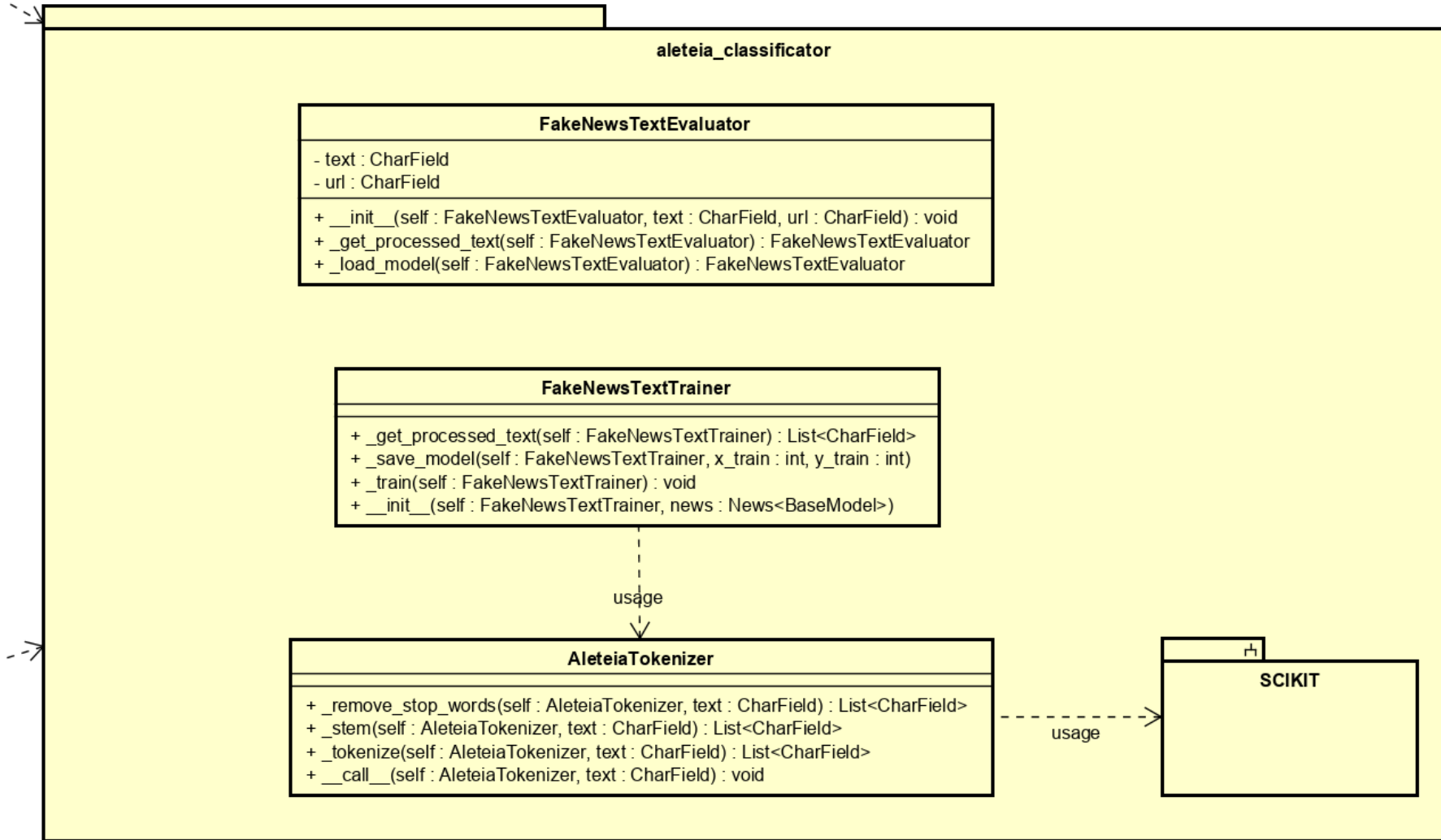
## APÊNDICE D - DIAGRAMA DE IMPLEMENTAÇÃO



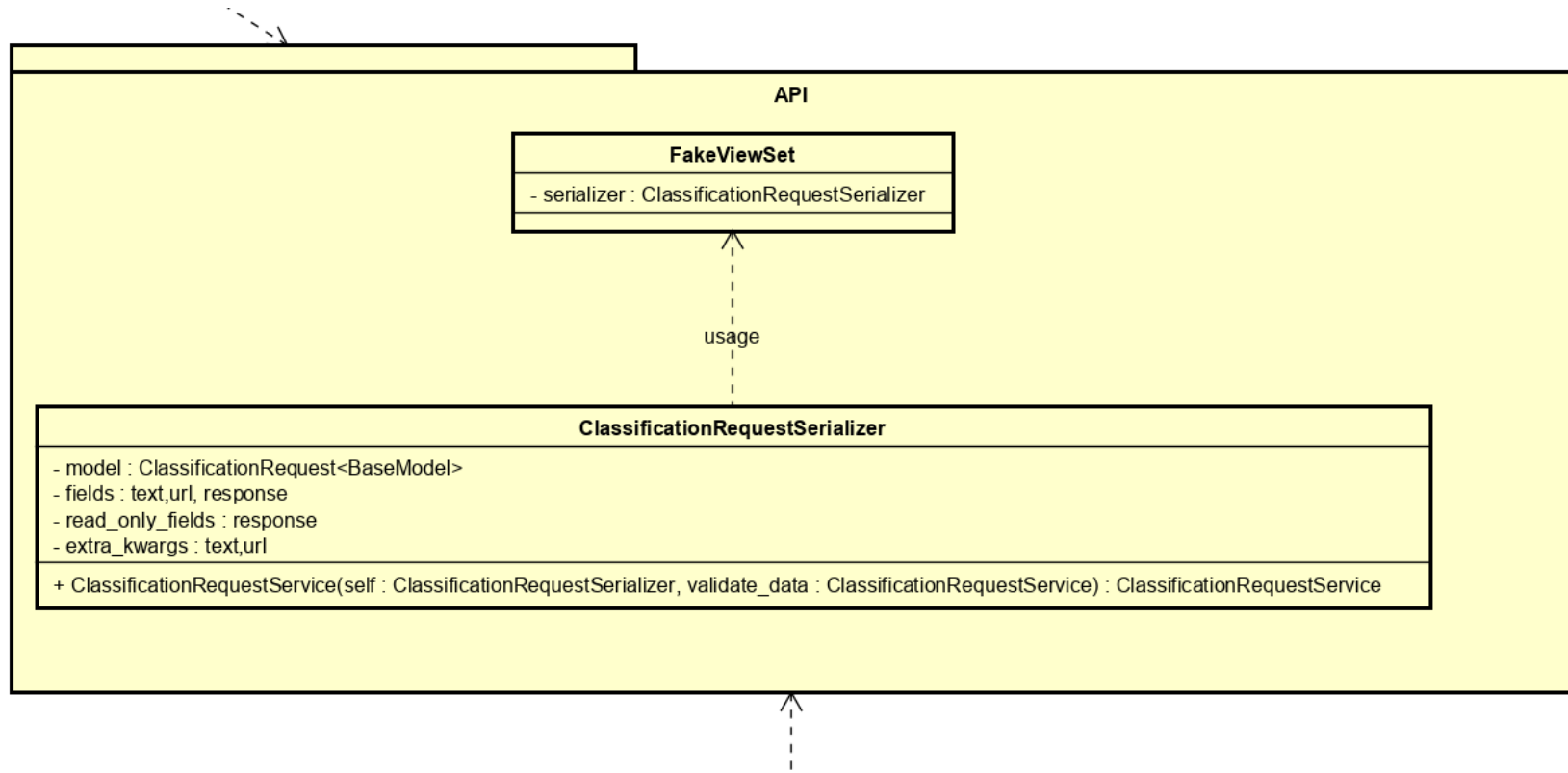
## APÊNDICE E - DIAGRAMA DE CLASSE: MODEL



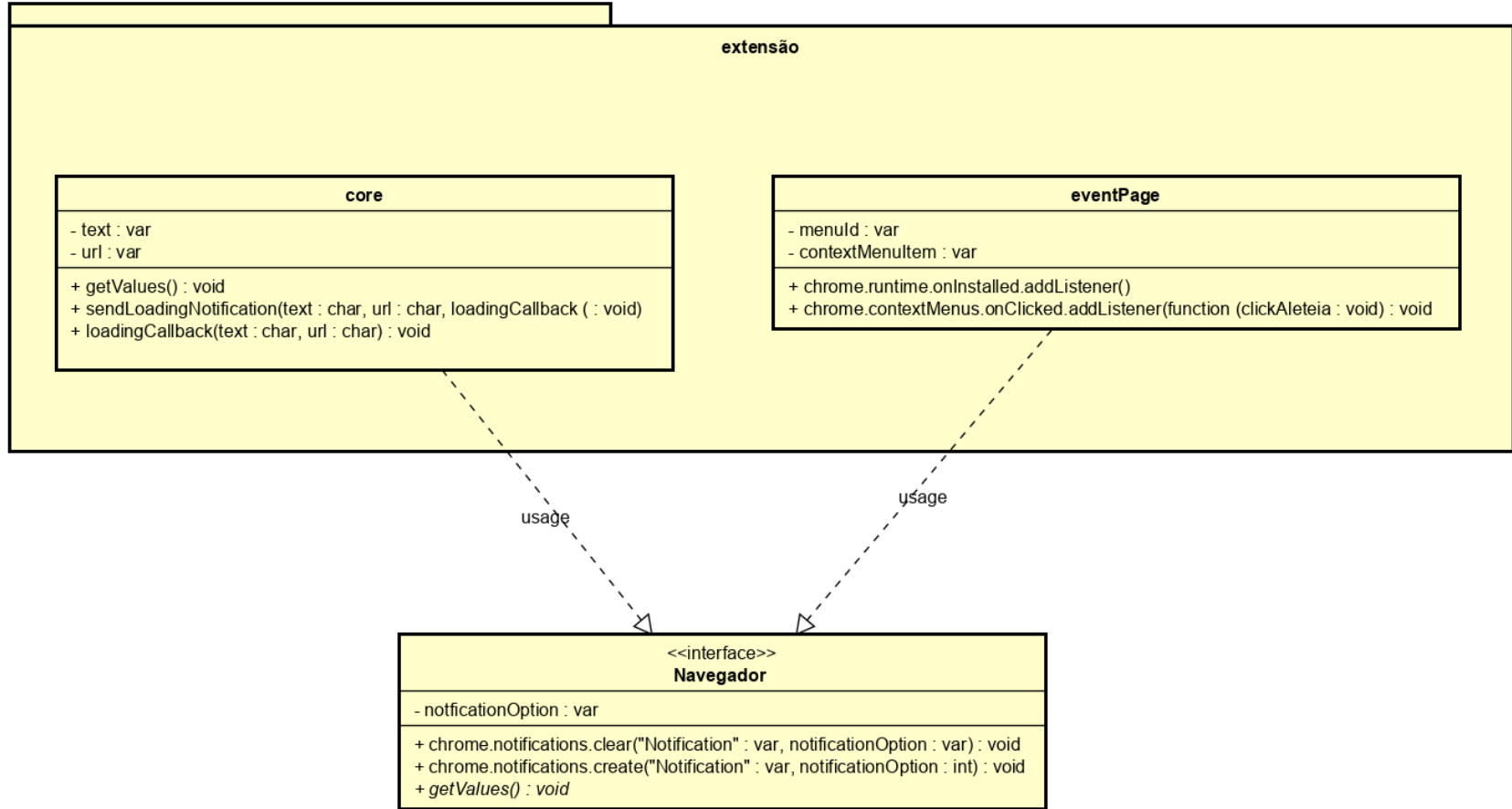
### APÊNDICE F - DIAGRAMA DE CLASSE: PACOTE ALETEIA\_CLASSIFICATOR



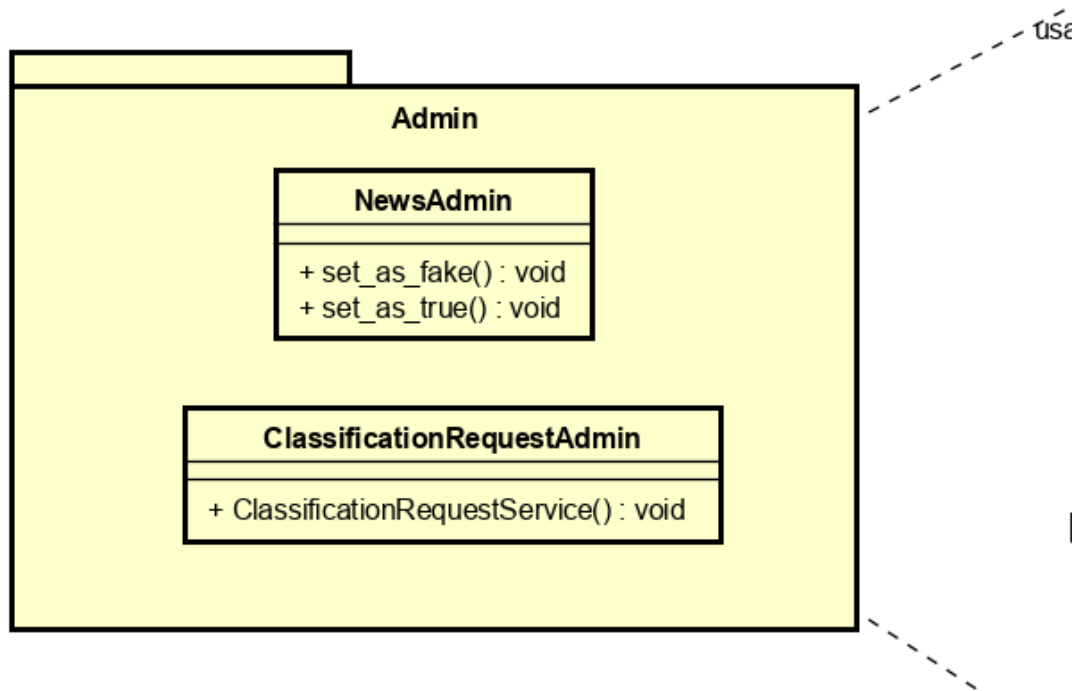
### APÊNDICE G - DIAGRAMA DE CLASSE: API



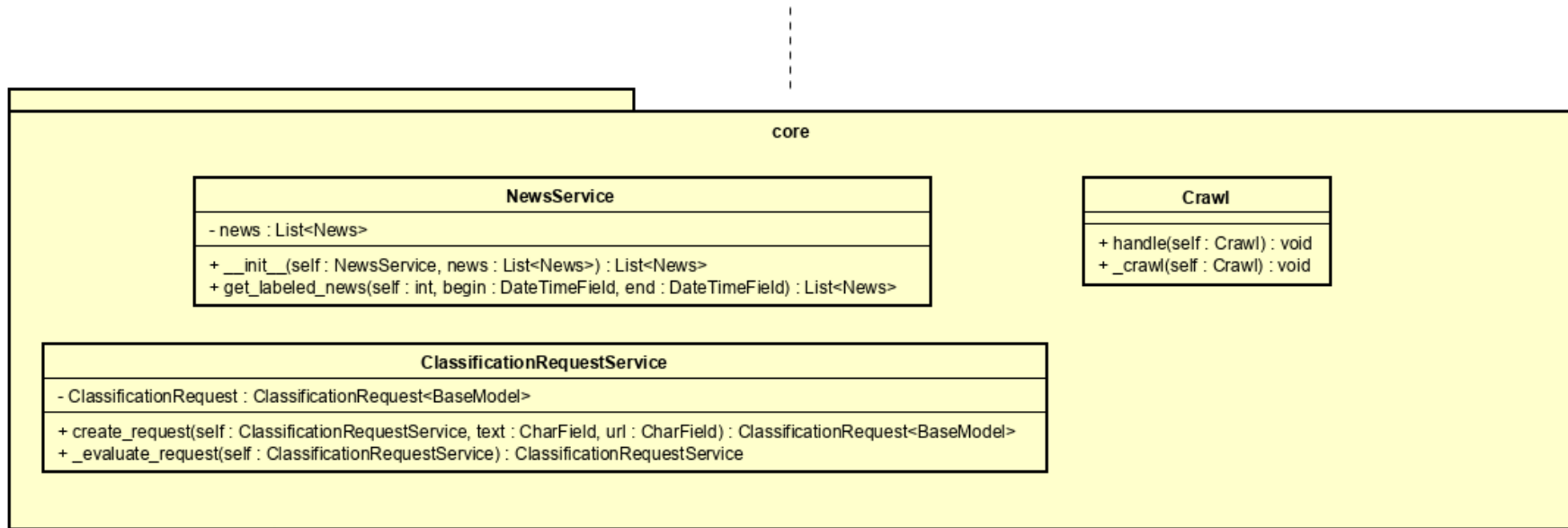
### APÊNDICE H - DIAGRAMA DE CLASSE: EXTENSÃO



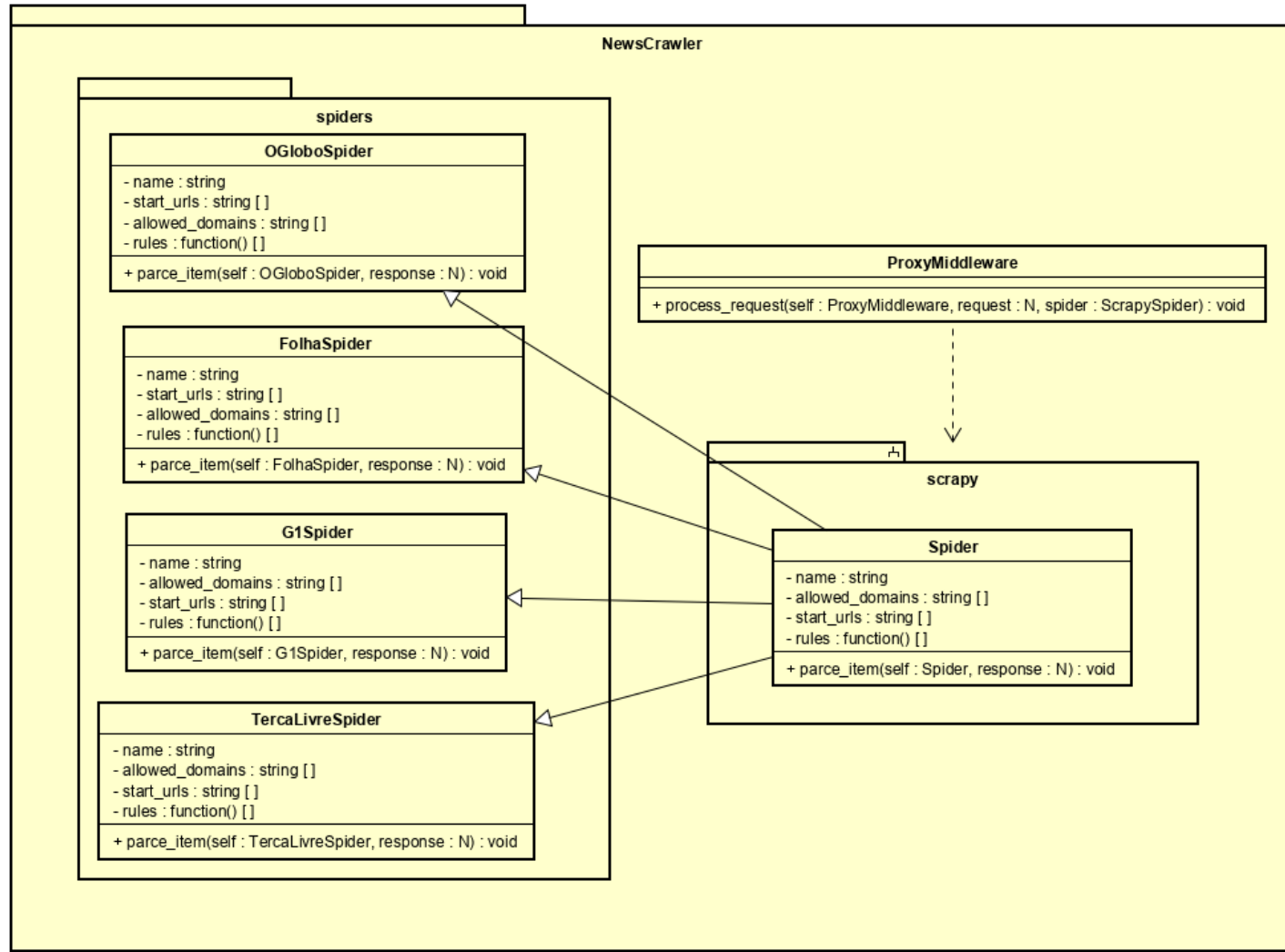
## APÊNDICE I - DIAGRAMA DE CLASSE: SERVICE ADMIN



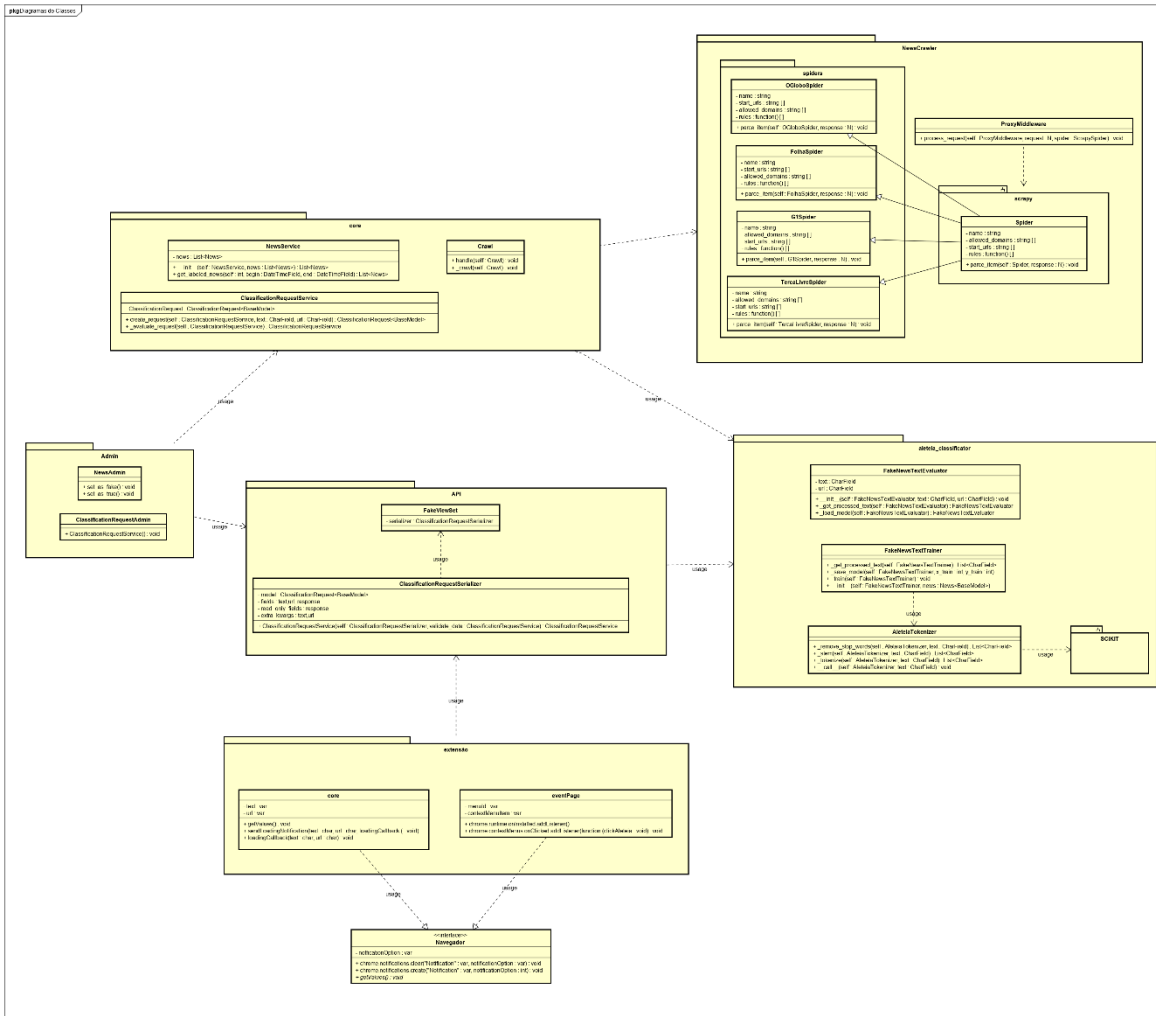
## APÊNDICE J - DIAGRAMA DE CLASSE: SERVICE CORE



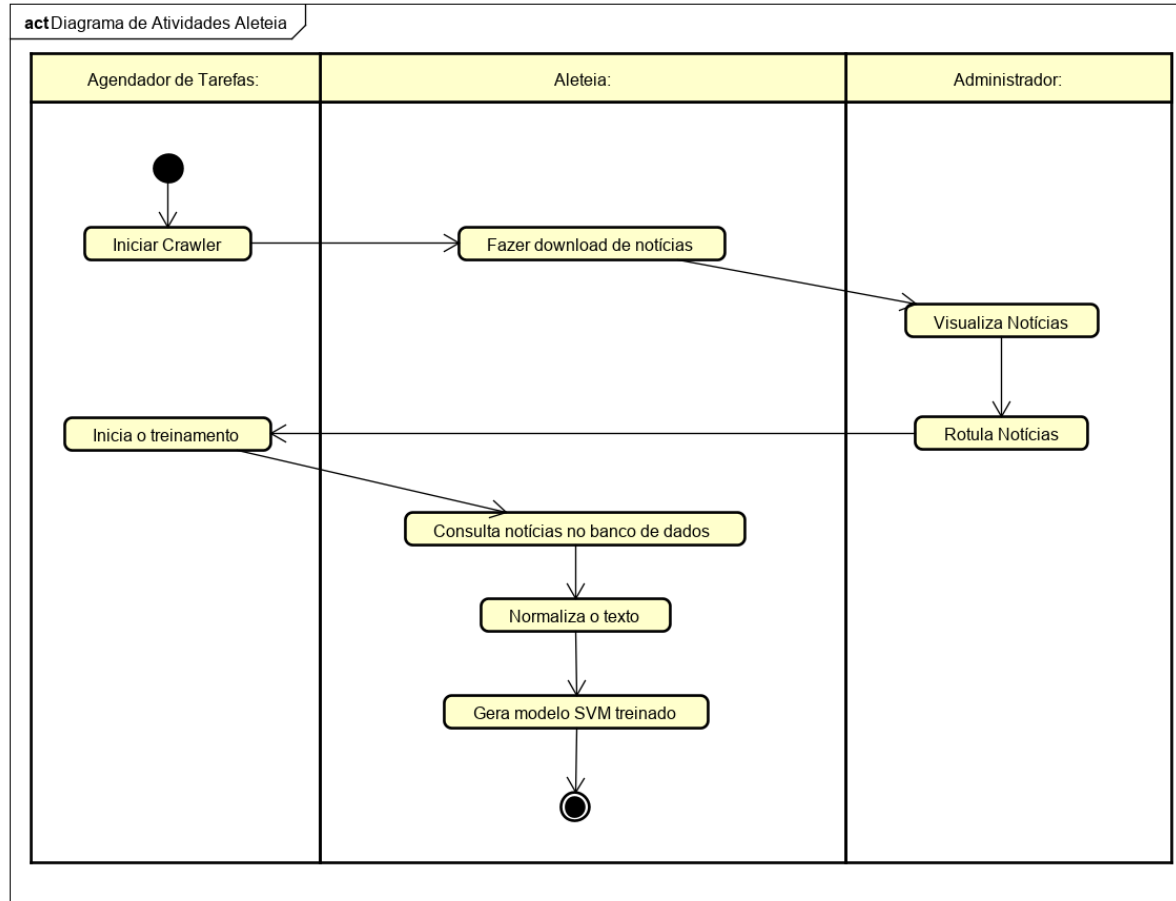
### APÊNDICE K - DIAGRAMA DE CLASSE: NEWSRAWLER



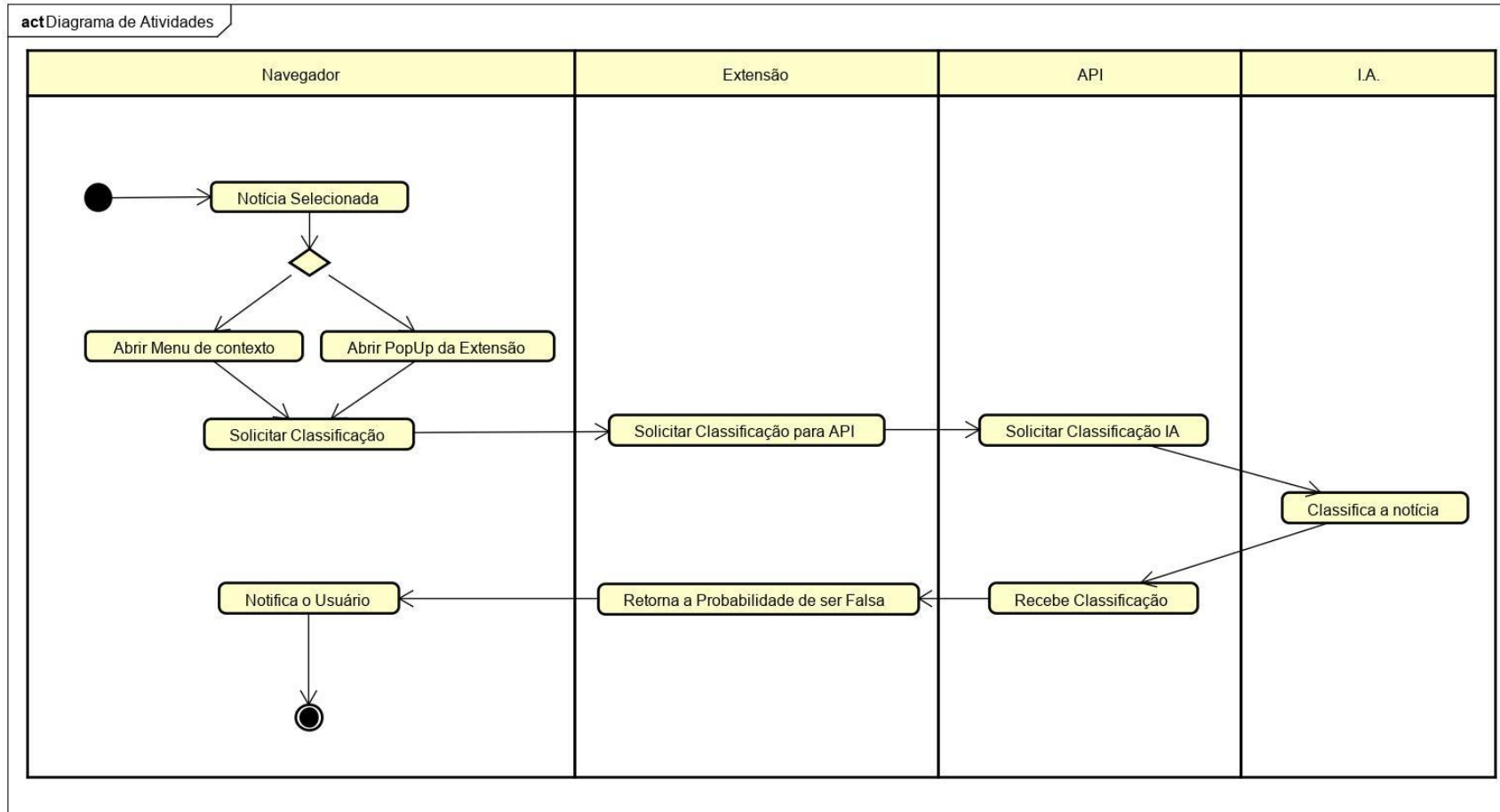
# APÊNDICE L - DIAGRAMA DE CLASSE: SERVICES



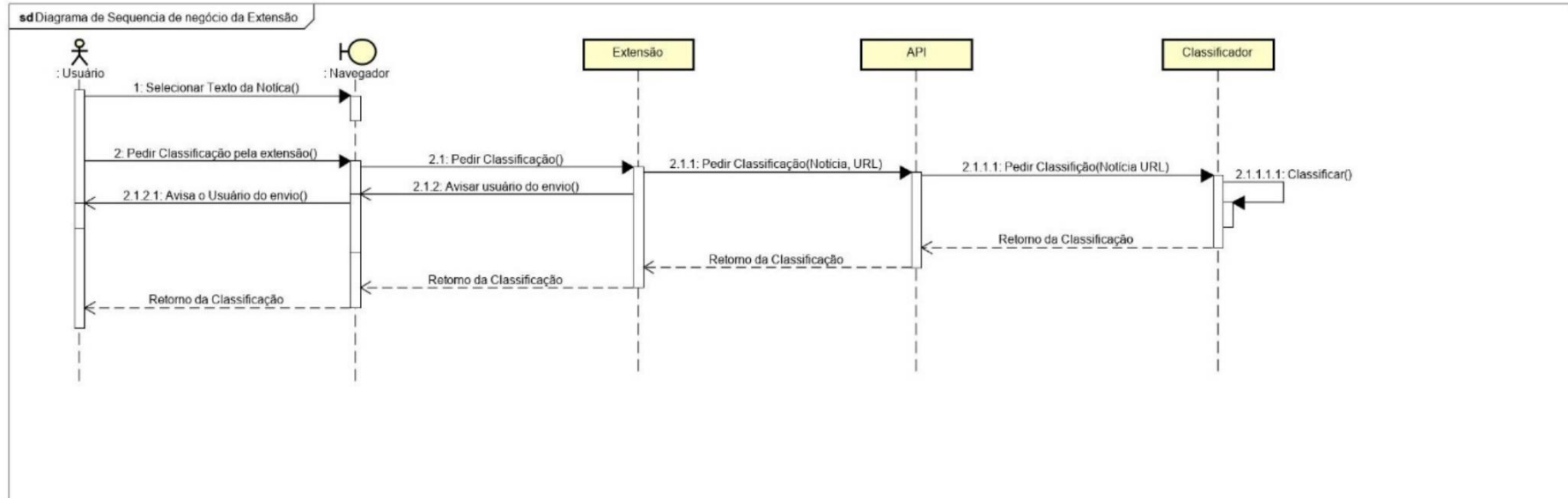
## APÊNDICE M - DIAGRAMA DE ATIVIDADES: ALETEIA



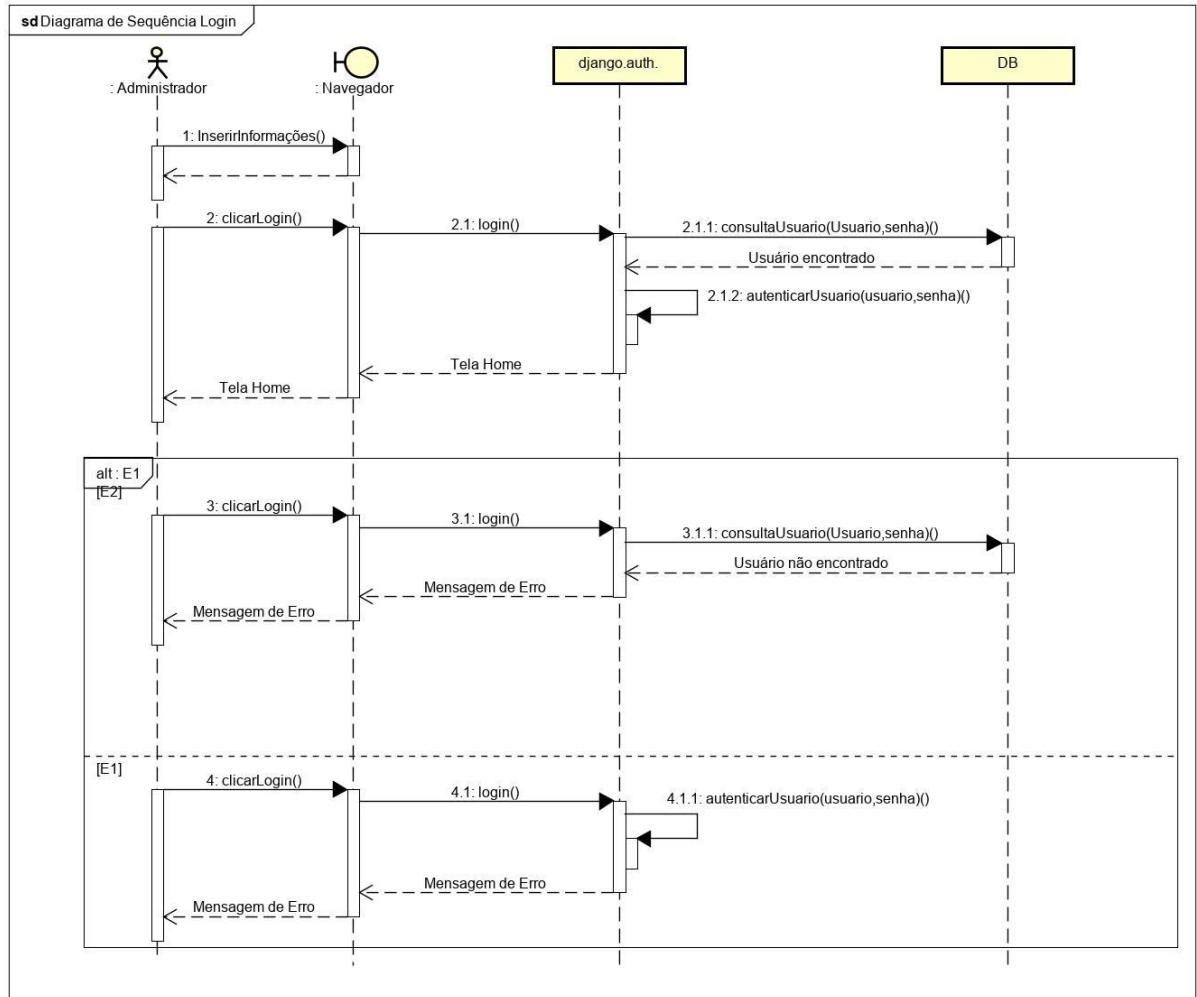
## APÊNDICE N - DIAGRAMA DE ATIVIDADES: EXTENSÃO



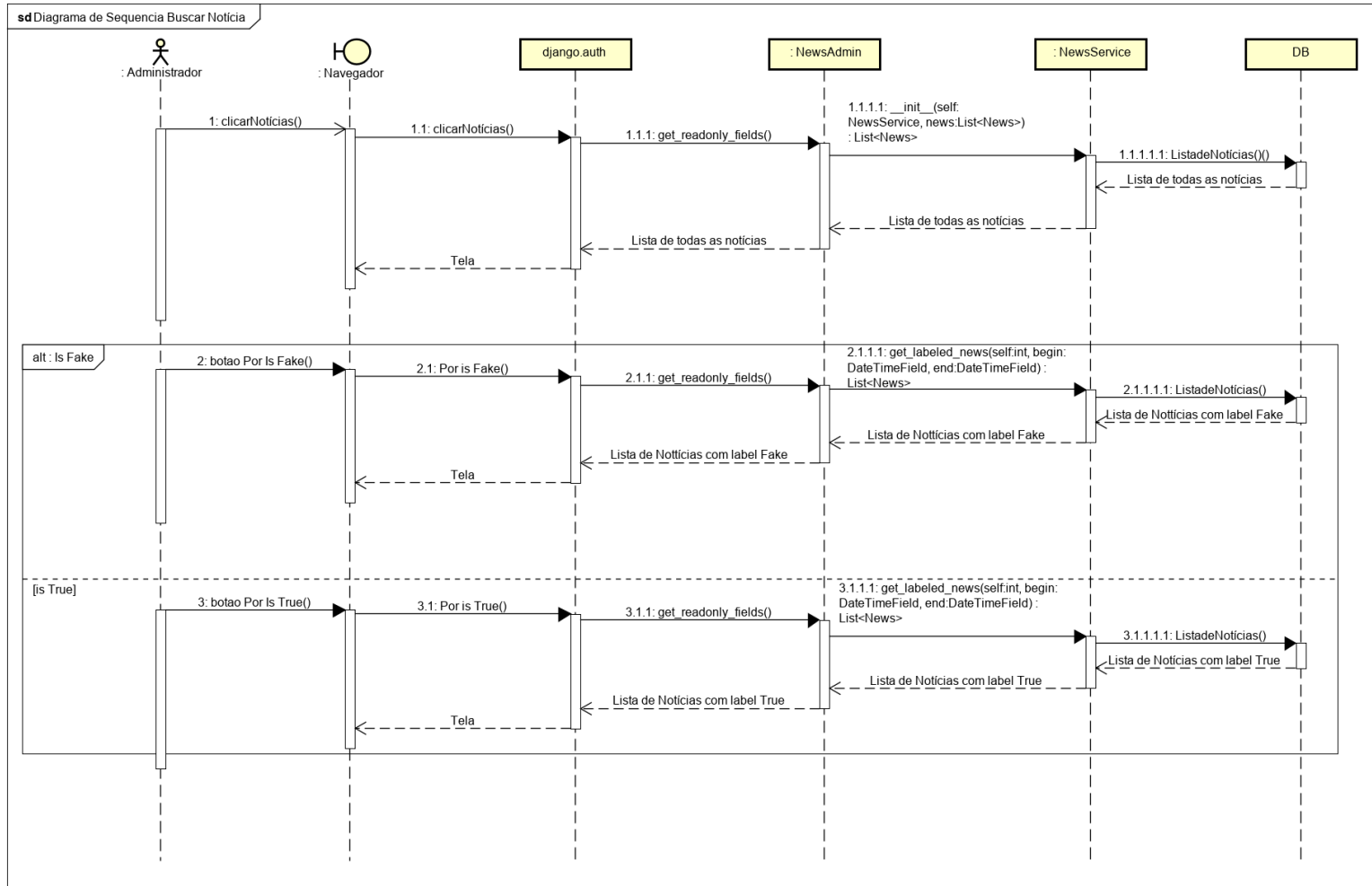
## APÊNDICE O - DIAGRAMA DE SEQUÊNCIA DE ANÁLISE: EXTENSÃO



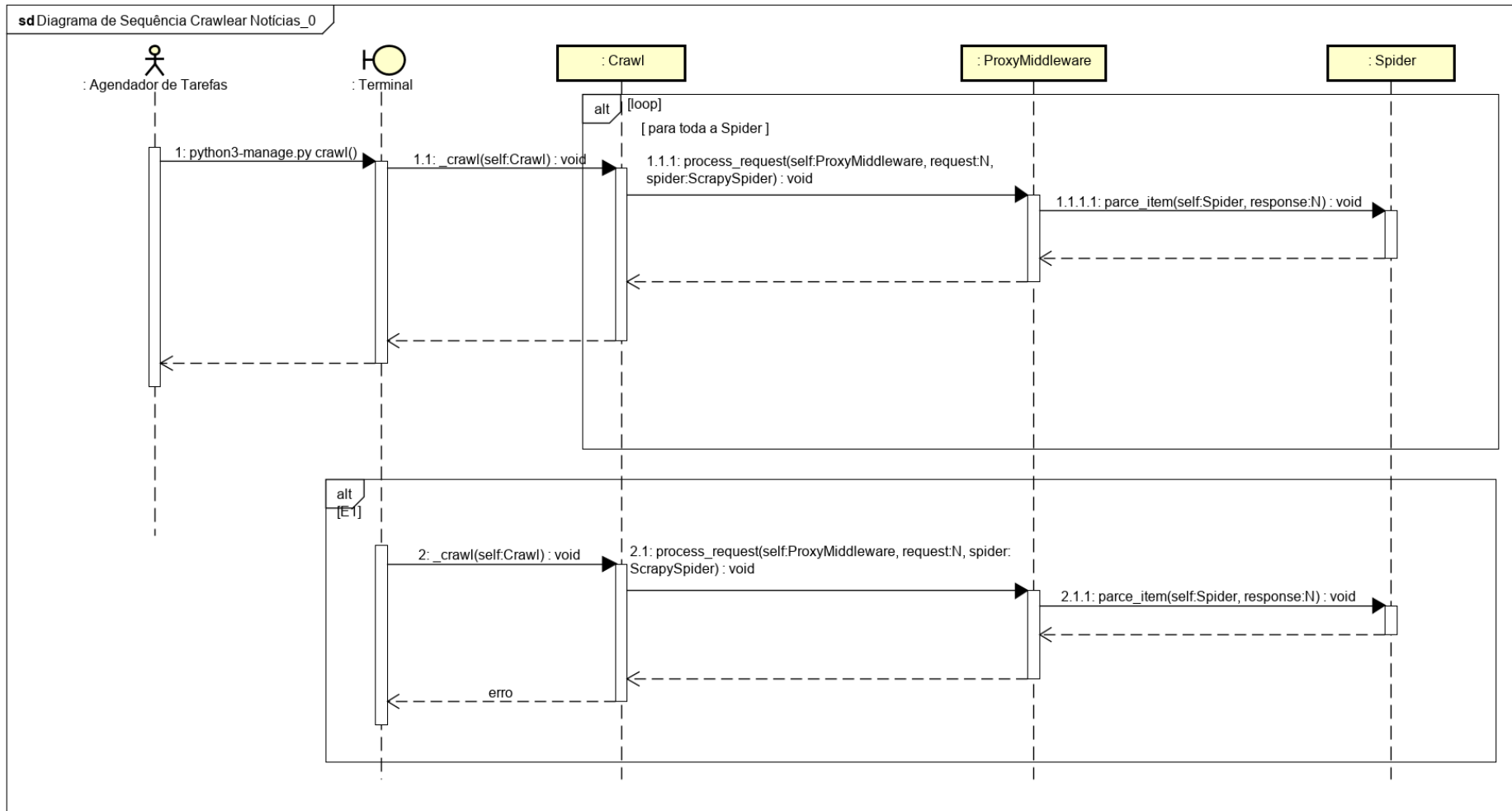
## APÊNDICE P - DIAGRAMA DE SEQUÊNCIA: LOGIN



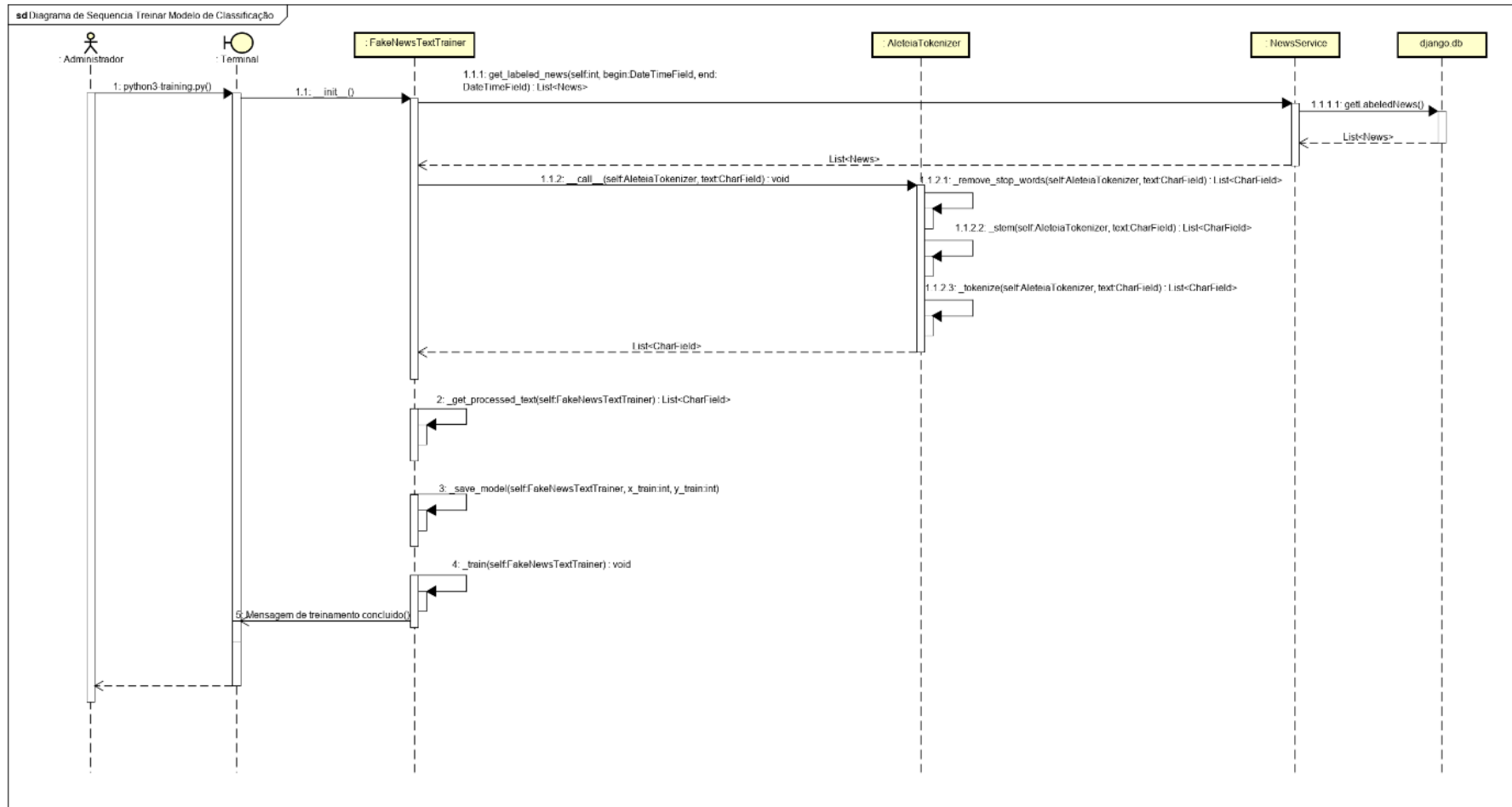
## APÊNDICE Q - DIAGRAMA DE SEQUÊNCIA: BUSCAR NOTÍCIA



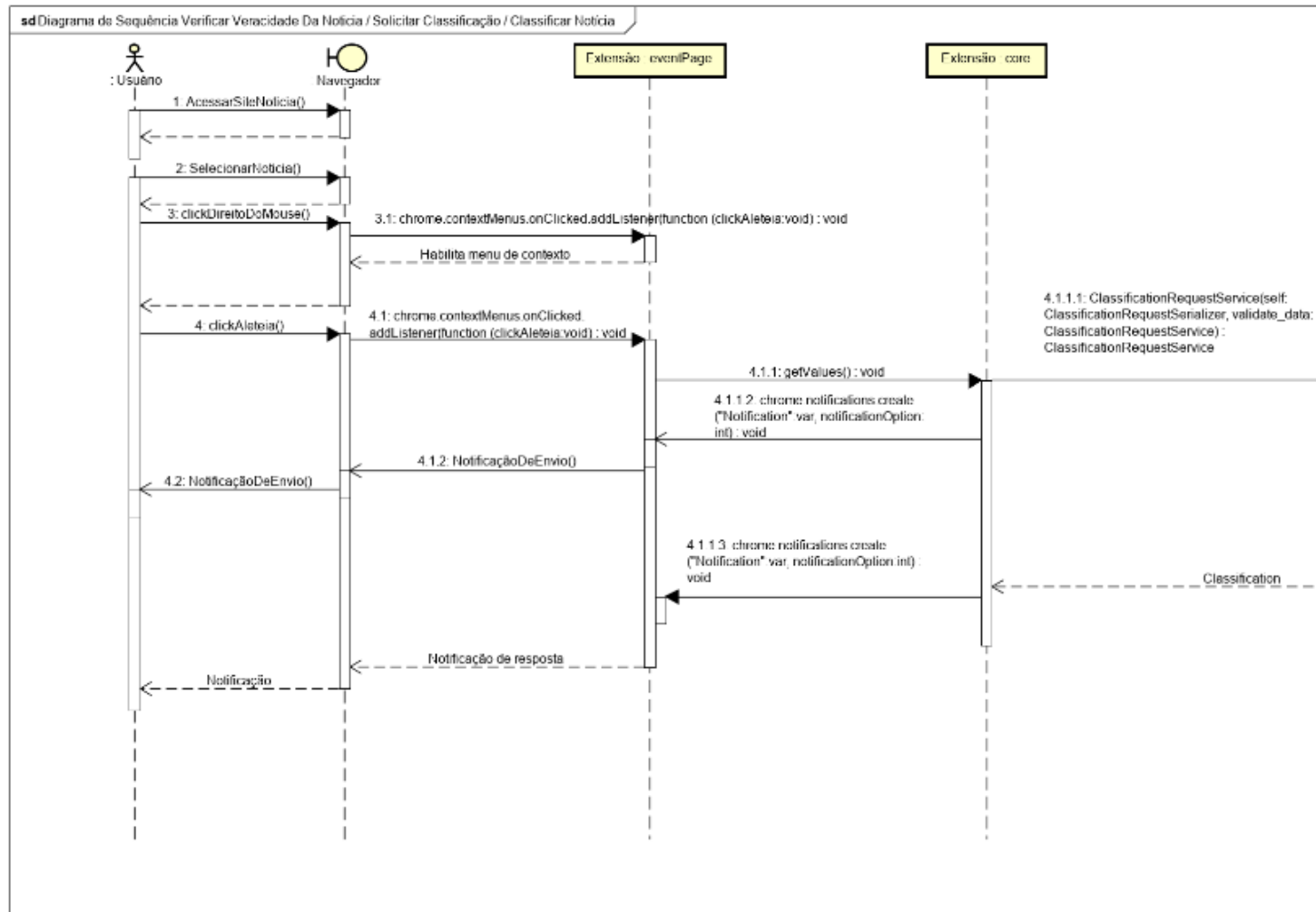
## APÊNDICE R - DIAGRAMA DE SEQUÊNCIA: CRAWLEAR NOTÍCIAS



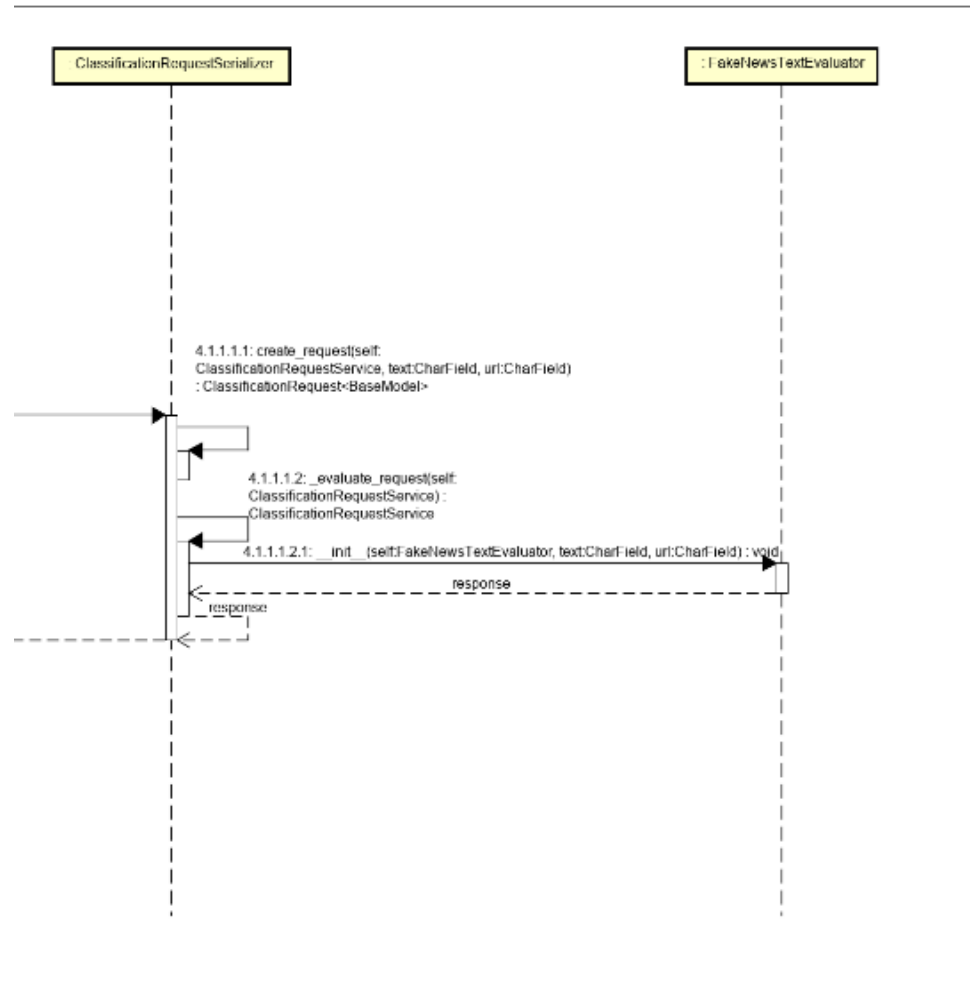
## APÊNDICE S - DIAGRAMA DE SEQUÊNCIA: TREINAR MODELO DE CLASSIFICAÇÃO



## APÊNDICE T - DIAGRAMA DE SEQUÊNCIA: VERIFICAR VERACIDADE DA NOTÍCIAS / SOLICITAR CLASSIFICAÇÃO / CLASSIFICAR NOTÍCIA - 1



## APÊNDICE T - DIAGRAMA DE SEQUÊNCIA: VERIFICAR VERACIDADE DA NOTÍCIAS / SOLICITAR CLASSIFICAÇÃO / CLASSIFICAR NOTÍCIA - 2



## APÊNDICE U - MODELO LÓGICO DE DADOS

core_news	
id	INT(11)
created_at	DATETIME(6)
updated_at	DATETIME(6)
date	DATETIME(6)
title	VARCHAR(255)
text	LONGTEXT
url	VARCHAR(2048)
is_fake	TINYINT(1)
classified_at	DATETIME(6)
Indexes	

core_classificationrequest	
id	INT(11)
created_at	DATETIME(6)
updated_at	DATETIME(6)
ipv4	VARCHAR(15)
ipv6	VARCHAR(40)
text	LONGTEXT
text_date	DATETIME(6)
url	VARCHAR(2048)
response	LONGTEXT
Indexes	

## APÊNDICE V - DIAGRAMA DE ARQUITETURA

