INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIAS E TECNOLOGIA DO AMAZONAS
CAMPUS MANAUS CENTRO
DEPARTAMENTO ACADÊMICO DE INFORMAÇÃO E COMUNICAÇÃO
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

PATRICIA CRISTINA SAMPAIO ARAGÃO

SISTEMA SEETER: ACOMPANHADOR DE COMENTÁRIOS DE SITES DE ACESSO PÚBLICO

MANAUS – AM
2021

PATRICIA CRISTINA SAMPAIO ARAGÃO

SISTEMA SEETER: ACOMPANHADOR DE COMENTÁRIOS DE SITES DE ACESSO PÚBLICO

Trabalho de Conclusão de Curso apresentado à banca examinadora Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistema do Instituto Federal de Educação, Ciências e Tecnologia do Amazonas – IFAM Campus Manaus - Centro, como requisito para o cumprimento da disciplina TCC II – Desenvolvimento de Software.

Orientador: Dr. Roceli Pereira Lima

MANAUS
2021

Biblioteca do IFAM – Campus Manaus Centro

A659s Aragão, Patrícia Cristina Sampaio.
Sistema SEETER: acompanhador de comentários de sites de acesso público / Patrícia Cristina Sampaio Aragão. – Manaus, 2021.
47 p. : il.

Trabalho de Conclusão de Curso (Tecnologia em Análise e Desenvolvimento de Sistema) – Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, *Campus* Manaus Centro, 2021.
Orientador: Prof. Dr. Roceli Pereira Lima.

1. Desenvolvimento de software. 2. Sites – acesso ao público. 3. Expressões regulares. I. Lima, Roceli Pereira. (Orient.) II. Instituto Federal de Educação, Ciência e Tecnologia do Amazonas III. Título.

CDD 005.3

Elaborada por Márcia Auzier CRB 11/597

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO MÉDIA E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA - AM
DEPARTAMENTO ACADÊMICO DE INFORMAÇÃO E COMUNICAÇÃO
CURSO DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA AMAZONAS

TERMO DE APROVAÇÃO

A monografia, que tem como título: **SETTEER – sistema acompanhador de comentários de sites de acesso público** foi submetida à defesa pública, sob a avaliação de banca examinadora, como parte dos requisitos necessários para a obtenção do título de graduação do curso superior de Tecnologia em Análise e Desenvolvimento de Sistemas.

AUTOR (A): PATRÍCIA CRISTINA SAMPAIO ARAGÃO

Monografia aprovada em: 18 / 08 / 2021

Orientador (a):

Primeiro (a) examinador (a):

JOAO GUILHERME DE MORAES SILVA:16016998249

Assinado de forma digital por JOAO GUILHERME DE MORAES SILVA:16016998249
DN: c=BR, o=ICP-Brasil, ou=Secretaria da Receita Federal do Brasil - RFB, ou=RFB e-CPF A3, ou=VALID, ou=AR ASCON, cn=JOAO GUILHERME DE MORAES SILVA:16016998249
Dados: 2021.08.19 19:51:41 -04'00'

Segundo (a) examinador (a):

AGRADECIMENTOS

Primeiramente a Deus e a Nossa Senhora, por terem traçado os melhores caminhos em minha vida e terem colocado sempre tudo em seu devido lugar.

A meus pais e familiares, por todo apoio, incentivo e compreensão durante o período de desenvolvimento deste Trabalho.

Aos meus professores do IFAM, por todos os ensinamentos compartilhados, de forma a me ajudar no processo de formação profissional. E especialmente ao Professor Doutor Roceli Lima, por ter sido um orientador muito presente e paciente.

E aos colegas do IFAM, que se tornaram amigos e que me ajudaram nos momentos de dificuldade nas disciplinas e marcaram minha história.

RESUMO:

A maioria dos sites, blogs e portais possuem uma seção de comentários onde é possível dar opiniões sobre o assunto, fazer perguntas, deixar elogios ou críticas e é de interesse dos usuários, algumas vezes, continuar acompanhando a atualização desses comentários. O sistema SEETER tem o objetivo de auxiliar esses usuários, realizando um acompanhamento de forma automatizada às seções de comentários dos sites que foram cadastrados no banco de dados do sistema. Para isso, foi realizada uma pesquisa para identificar expressões regulares que correspondem aos comentários de diversos sites e utilizou-se o Web Scraping para realizar a contagem constante dos comentários dos links cadastrados. A partir disso, foi desenvolvido um sistema com Back-End e Front-End integrados capaz de verificar e notificar os usuários a cada vez que a quantidade de comentários for alterada. Dessa forma, os usuários podem registrar diversos links de sites no SEETER e deixar que o próprio sistema realize o monitoramento e os notifique quando novos comentários forem publicados.

Palavras-Chave: Comentários. Sites de Acesso Público. Expressões Regulares. Web Scraping.

ABSTRACT:

Most websites, blogs and portals have a comments section where it is possible to give opinions about the subject, ask questions, leave compliments or criticisms, and it is in the users' interest, sometimes, to continue following the update of these comments. The SEETER system is intended to help these users by monitoring in an automated way the comments sections of the sites that have been registered in the system's database. To do this, a search was done to identify regular expressions that correspond to the comments from different sites, and Web Scraping was used to perform a constant count of the comments from the registered links. From this, a system was developed with integrated Back-End and Front-End, that is able to check and notify users each time the number of comments changes. In this way, users can register several site links in SEETER and let the system do the monitoring and notify them when new comments are posted.

Keywords: Comments. Public Access Sites. Regular Expressions. Web Scraping.

LISTA DE FIGURAS

SUMÁRIO

1 INTRODUÇÃO

A evolução do uso da Internet é notória nos últimos anos. A qualquer hora do dia em qualquer lugar do mundo existem milhares de pessoas conectadas à Internet. Não é preciso ir tão longe para encontrar alguém acessando notícias em um site, conversando virtualmente pelas redes sociais, jogando, ouvindo música, assistindo filmes, palestrando, trabalhando remotamente, realizando compras de produtos, etc.

Ao longo do tempo, as pessoas conseguiram ter mais acesso à Internet. As estatísticas da página Internet World Stats mostram que até 31 de Dezembro de 2019 eram mais de 4,5 bilhões de pessoas usuárias da Internet (Internet World Stats, 2020). De acordo com dados da pesquisa TIC Domicílios, realizada em 2018, 70% dos brasileiros são usuários da Internet, o que significa quase 126,9 milhões de pessoas conectadas (TIC Domicílios, 2018).

Pesquisas revelam também que de 2010 a 2019, o número de usuários da Internet cresceu cerca de 1.114% (Internet World Stats, apud Olhar Digital, 2019). Alguns dos fatores que contribuíram para essa elevada porcentagem foram os acessos mais facilitados e acessíveis quanto ao valor à Internet através dos Smartphones e o valor dos serviços de Internet, que se tornaram mais justos e adequados ao orçamento dos brasileiros.

Sendo assim, é possível encontrar diversos sites que abordam temas variados, como Economia, Política, Religião, Cultura e Comportamentos. E ainda existem aqueles que apresentam diferentes pontos de vista sobre assuntos da atualidade, é o caso dos sites de jornalismo independente. Essa enorme quantidade de atividades na Web trouxe muitas informações para a Internet, gerando um verdadeiro Big Data.

Segundo Enomura (2014, apud PATRICIO, 2018), Big Data é o termo utilizado para representar um banco de dados extremamente amplo que necessita de estudos e formas inovadoras de processamento capazes de analisar essa grande quantidade de dados e melhorar a percepção e tomadas de decisões.

Por outro lado, é possível notar também que o desenvolvimento de novas linguagens de programação, novos frameworks e novas tecnologias também ajudaram no desenvolvimento das páginas Web e na melhoria dos sites. Páginas que antes eram feitas utilizando apenas textos e hipertextos, hoje

contam com a ajuda de imagens e vídeos, elas são mais interativas e oferecem ao usuário uma melhor experiência.

Segundo dados da Netcraft, a quantidade de sites ativos até abril de 2019 era de aproximadamente 1.45 bilhão (Netcraft, 2019). Hoje, existem sites de notícias, blogs, fóruns, sistemas acadêmicos, redes sociais, sem falar dos inúmeros aplicativos para pedir comida, de bate-papo e outros serviços. Com tantos sites e aplicativos, acompanhar uma página, às vezes se torna uma tarefa difícil, visto que elas sofrem atualizações.

Para auxiliar os usuários da Internet em meio ao Big Data, várias ferramentas foram produzidas. A pesquisa no Google é uma das formas mais conhecidas para realizar buscas pela Internet, e ainda assim, a plataforma dispõe de outras ferramentas para melhorar a sua precisão com a seleção do idioma e de intervalos de datas. Ainda é possível contar com o Google Acadêmico e o Google Advanced Search para pesquisas em artigos e pesquisas avançadas. Existem também outros sites como o DuckDuckGo e StartPage, que também são ferramentas de busca, mas estas últimas garantem uma melhor privacidade.

Um exemplo de pesquisa no Big Data ocorre quando o internauta deseja procurar algo de seu interesse. Quando encontra uma notícia ou página que corresponda a sua busca, algumas vezes, pode ser de seu interesse continuar acompanhando a discussão do tema e para isso, precisar realizar leitura constante dos comentários. Um exemplo: a atual situação da pandemia do Covid-19 está gerando uma enorme quantidade de informações a cada hora do dia, em diversos sites. Para que os usuários consigam acompanhar os comentários sobre o assunto, em determinada página, atualmente, eles precisam visitar a página web várias vezes no dia e verificar se existem novos comentários, demandando um grande esforço de tempo.

A proposta desse projeto de conclusão de curso é desenvolver um software para auxiliar o usuário da Internet a acompanhar os comentários publicados em páginas web. A intenção é projetar um artefato de software que a partir da escolha de alguma notícia publicada na Internet (ver, to see), este sistema possa “olhar” e avisar o internauta quando houver uma nova publicação de comentário.

1.1 PROBLEMATIZAÇÃO

Com a ampla quantidade de dados gerados na Internet diariamente, e com o grande número de sites, blogs, redes sociais, aplicativos e outros sistemas que são possíveis de acessar na Internet, a tarefa de acompanhar uma notícia ou informação de interesses se torna difícil, ainda mais se feita manualmente.

Acompanhar os comentários de sites de acesso público pode ser uma atividade que demanda bastante tempo dos usuários, se feito de forma manual.

Sendo assim, de que forma um sistema de software pode ajudar usuários com interesse em acompanhar comentários de sites específicos? Qual público tem interesse nesse tipo de sistema? Há mais alguma vantagem em utilizar um software desse tipo além da vantagem de tempo? Já existem softwares com funcionalidades similares no mercado?

1.2 JUSTIFICATIVA

A escolha do tema do projeto “SISTEMA SEETER – acompanhador de comentários de sites de acesso público” foi uma sugestão do orientador Professor Dr. Roceli Lima. A justificativa que levou ambos a considerarem o tema relevante foi o fato de atualmente não existirem plataformas que realizem a função de automatizar e concentrar, num único lugar, notificações de comentários de sites de interesse do usuário. No momento, o usuário que tiver a necessidade de acompanhar a seção de comentários de um determinado site, precisará fazer o acompanhamento de forma manual, demandando tempo para realizar uma tarefa, e que poderia ser simples.

Outro fator que também indica que o projeto poderia trazer vantagem aos seus usuários é a comparação com outras redes sociais, como o Facebook e o Instagram. Nesses aplicativos, os usuários podem selecionar pessoas ou instituições às quais querem acompanhar suas postagens. Assim também, com o uso do SEETER, usuários poderão selecionar apenas os sites de seus interesses para continuarem acompanhando os comentários.

Vale ainda citar que a ferramenta pode ser um auxílio para equipes de assessoria de comunicação de empresas, instituições ou até mesmo de famosos, que necessitam todos os dias acompanhar a repercussão dos assessorados em sites e blogs.

Esse sistema irá unir todos os sites de interesse do usuário e irá notificá-lo sempre que houver atualizações na quantidade de comentários. O sistema ajudará assim empresas ou instituições que, caso sejam citadas em alguma página web, poderão acompanhar os comentários a respeito delas, de maneira rápida e fácil e todos os seus usuários a acompanharem os sites de seus interesses.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

Desenvolver um sistema para o usuário acompanhar comentários postados em sites de acesso público, a fim de notificá-lo sempre que houver uma atualização na seção de comentários, sendo isso possível através da verificação constante do sistema à quantidade de comentários.

1.3.2 Objetivos Específicos

- Propor uma arquitetura com os principais componentes de um sistema para armazenar dados de sites de acesso público e seus respectivos comentários registrados.
- Desenvolver aplicação com Back-end e Front-end integrados de forma que permita conexão com banco de dados, lógica do negócio e interface para os usuários.
- Projetar e desenvolver o Sistema SEETER para acesso aos comentários publicados em sites de acesso público, a fim de notificar o usuário sempre que a quantidade de comentários for atualizada em sites registrados.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 INTRODUÇÃO

A grande capacidade humana em gerar novas informações era muito maior do que a capacidade em armazená-la de forma a permitir futuras consultas, era o que afirmava Vannevar Bush (1945, apud CRISTOVÃO, 2016). Ele já alertava para a necessidade de melhorar a recuperação de informações e conhecimento.

O Sistema SEETER visa permitir que seus usuários acompanhem sites de acesso público através da seção de comentários. Pretende ainda, notificá-los sempre que houver alteração na quantidade de comentários dos sites selecionados.

Para o desenvolvimento deste Trabalho de Conclusão de Curso foram realizadas pesquisas sobre: Máquina de Busca, Recuperação de Informações e Expressões regulares. Estes são temas que serão abordados neste capítulo. Também serão citadas tecnologias e ferramentas que auxiliarão o desenvolvimento deste projeto: SCRUM, Web Scraping e Web Crawler, além das tecnologias utilizadas: Python, Django Rest Framework e Framework Angular.

2.2 MÁQUINA DE BUSCA

Segundo Caio Daoud (2016), as Máquinas de Busca são sistemas que procuram informações relevantes em uma coleção de documentos presentes na Internet, de acordo com a necessidade do usuário. Ou seja, as Máquinas de Busca vão realizar uma consulta de acordo com as palavras-chave (ou termos) fornecidas pelo internauta. O objetivo da busca é apresentar uma lista de documentos de relevância à pesquisa do usuário.

> As Máquinas de Busca mudaram o modo de acesso das pessoas à informação, tornando-o fácil, de modo que qualquer um pode obter informações rapidamente, digitando poucas palavras em um campo de pesquisa. (CAMBAZOGLU; BAEZA-YATES, 2015 apud GAIOSO, 2019, p. 18).

As Máquinas de Busca funcionam por meio dos índices invertidos, que são estruturas de dados que organizam as informações dos documentos resultantes das consultas. Essas estruturas também são constituídas por vocabulários e listas invertidas, as quais contém a quantidade de vezes que os termos aparecem em cada documento. (GAIOSO, 2019).

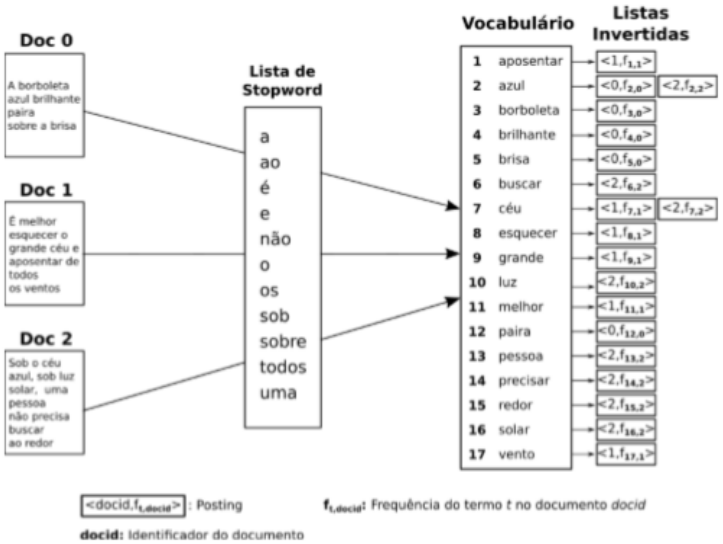
E para que os resultados das Máquinas de Busca sejam eficientes, um dos critérios é de que elas encontrem documentos que correspondam às necessidades expressas nas consultas dos usuários, em um curto espaço de tempo. (GAIOSO, 2019).

De acordo com Roussian Gaioso (2019), as Máquinas de Busca localizam em listas invertidas os termos da consulta, para que possam encontrar os documentos resultantes de determinada consulta.

As duas técnicas principais que são utilizadas pelas Máquinas de Busca são: Document-At-A-Time (DAAT) e Term-At-A-Time (TAAT). Em DAAT, as listas são atravessadas simultaneamente, enquanto em TAAT somente uma lista invertida é processada por vez. (GAIOSO, 2019, p. 18).

O índice invertido será essencial no momento da busca dos documentos que vão compor a lista resultante da consulta do usuário. Ele irá mapear cada termo relevante que existe em uma coleção de documentos. A partir disso, duas estruturas são formadas: o vocabulário e as listas invertidas (GAIOSO, 2019).

Figura 1 - Índice Invertido



Fonte: GAIOSO (2019).

A Figura 1 representa um índice invertido, um vocabulário e listas invertidas a partir de uma coleção com três documentos.

> Observa-se que uma lista de stopwords é utilizada para remover do vocabulário as palavras extremamente comuns e que pouco contribuem para selecionar os documentos de acordo com a necessidade dos usuários. O uso de listas de stopwords reduz drasticamente o tamanho das listas invertidas (MANNING et al., 2008 apud GAIOSO, 2019, p. 25).

Após o entendimento sobre o funcionamento das Máquinas de Busca, concluiu-se que esse recurso não será utilizado neste projeto, por conta de o objetivo dessa ferramenta ser uma busca por documentos e o objetivo do projeto trabalhar com páginas Web já anteriormente escolhidas pelo usuário.

2.3 RECUPERAÇÃO DE INFORMAÇÃO

De acordo Liu (2006, apud CRISTOVÃO, 2016), o estudo da Recuperação de Informação tradicional foca na relação entre uma consulta textual e os documentos da base de conhecimento, com o intuito de gerar rankings com os documentos que forem recuperados.

Para apresentar os resultados relevantes de uma determinada consulta, geralmente é utilizado um modelo de Recuperação de Informação, como por exemplo, o Modelo de Espaço Vetorial ou o BM25. Esses modelos servem para calcular os escores de cada documento e ordená-los. Escore é o grau de similaridade entre um documento e a consulta, então ele pode ser considerado uma estimativa da relevância do documento como resposta à consulta. Os documentos que resultarem em maiores escores serão selecionados para compor a lista de resultados que será retornada ao usuário ou que será utilizada por outro processo de ordenação mais complexo (DAOUD, 2016).

O Modelo de Espaço Vetorial é um dos modelos mais tradicionais na área da Recuperação de Informação. Esse modelo foi proposto por Salton et al e funciona da seguinte forma:

> [...] os documentos são representados como vetores, e o tamanho do vocabulário é a dimensão dos vetores. Dois documentos têm seu valor de similaridade determinado pela fórmula do cosseno, onde documentos idênticos têm

> valor de similaridade 1 e documentos completamente diferentes de similaridade 0. (DAOUD, 2016, p. 5).

Já o BM25 é outro modelo de similaridade, de Robertson et al, cujo resultado deste modelo é o resultado de vários experimentos e variações aplicadas a um clássico modelo probabilístico (DAOUD, 2016).

Esses dois modelos são aplicados, em uma primeira fase de consulta, a todos os documentos das listas invertidas, a fim de formar uma lista com os documentos de maior relevância. Já na segunda fase, “é utilizada uma função de ordenação baseada em aprendizado de máquina no conjunto de documentos candidatos obtidos na primeira fase, gerando uma nova ordenação final que é apresentada ao usuário” (GAIOSO, 2019, p. 20).

> A Figura 2 representa um Sistema de Recuperação de Informação que é composto pelos seguintes itens: documentos, necessidades do usuário que geram a formulação de consultas, e finalmente, a Recuperação da Informação, que depende do alinhamento entre o processo de indexação dos documentos e a busca realizada. Como produto disso, uma lista de documentos considerados relevantes é apresentada ao usuário solicitante. Havendo significativa divergência entre os termos de indexação e a busca dos usuários, as possibilidades de perda de informações são alavancadas, e o processo de Recuperação da Informação tende a ser menos eficaz. (SOBRAL, 2018, p. 42).

Figura 2 - Itens de um Sistema de Recuperação de Informação

Fonte: GEY (1992, apud SOBRAL, 2018).

A ideia inicial seria de que a Recuperação de Informação ao invés de gerar ranking de documentos de acordo com o grau de similaridade, a contagem dos índices seria utilizada apenas para a verificação da quantidade de comentários de determinada página Web, o que permitiria saber se houve a publicação de um novo comentário. Porém, após aprofundar o entendimento sobre a ferramenta, constatou-se que neste projeto não será utilizada a ferramenta de Recuperação de Informação.

Será necessário fazer uma pesquisa e identificar as principais expressões que correspondem à seção de comentários e estabelecer uma lista de expressões regulares para que o sistema, de fato, possa encontrar a seção e conseguir realizar suas atividades.

2.4 EXPRESSÕES REGULARES

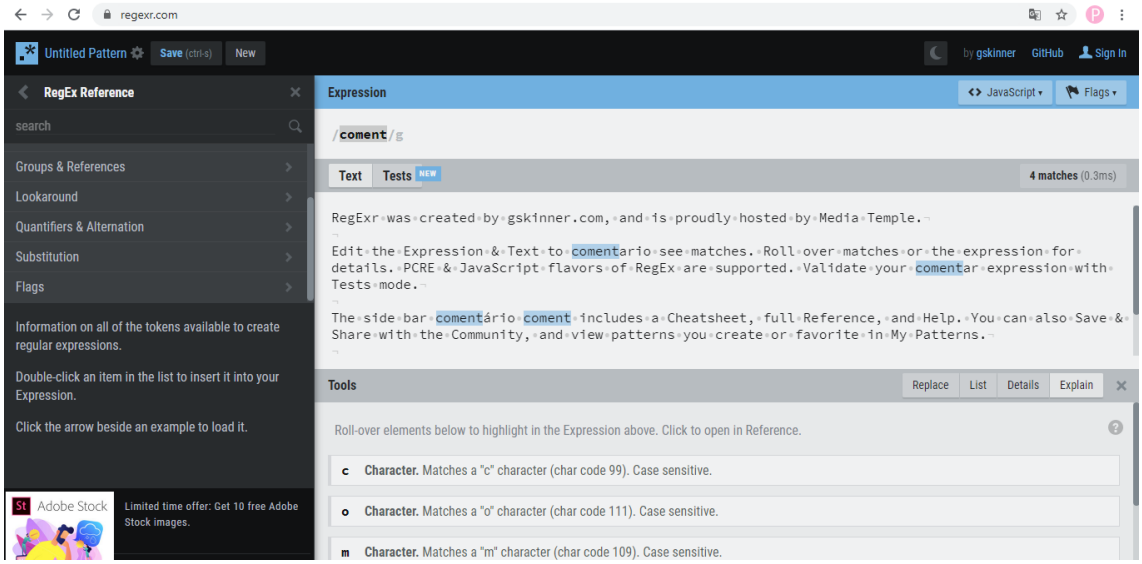
Para o desenvolvimento do Sistema SEETER, será necessário encontrar, em cada site de acesso público selecionado pelos usuários, a seção de comentários para que o sistema faça a contagem da quantidade de comentários, registre a quantidade atual e notifique o usuário caso a quantidade aumente. Para isso, serão identificadas expressões regulares que permitam localizar a seção.

> Uma expressão regular é um método formal de se especificar um padrão de texto, sendo composta por símbolos e caracteres com funções especiais, que quando agrupados formam uma sequência/expressão (JARGAS, 2016 apud INÁCIO, 2020, p. 8).

Ao criar uma expressão regular, será possível utilizá-la para encontrar em textos ou códigos, palavras e caracteres que correspondam à expressão, sendo interpretada como uma regra que quando uma entrada de dados se associar com essa regra, ou seja, obedecer exatamente a todas as condições, é retornado o dado que atende as mesmas. (INÁCIO, 2020, p. 8).

A Figura 3 mostra um exemplo de uma expressão regular para encontrar em um texto, palavras que se aproximem a “coment”, usando a ferramenta RegExr.

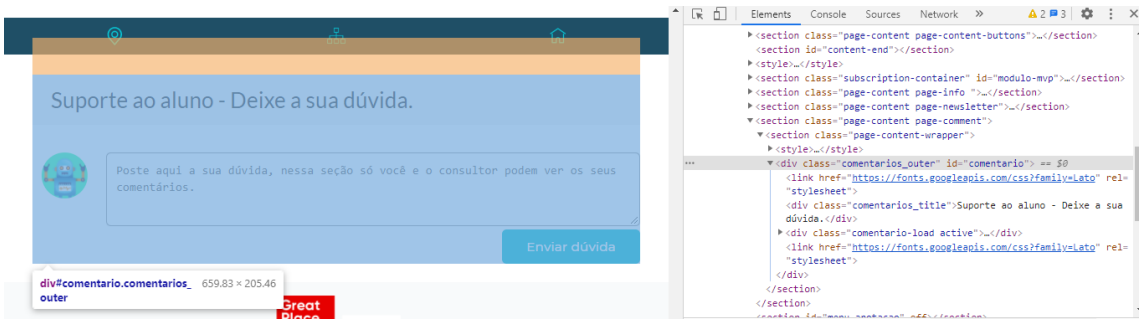
Figura 3 - Uso de Expressões Regulares



Fonte: Próprio autor.

As Figuras 4 e 5 abaixo representam as várias expressões regulares que são utilizadas para representar a seção de comentários.

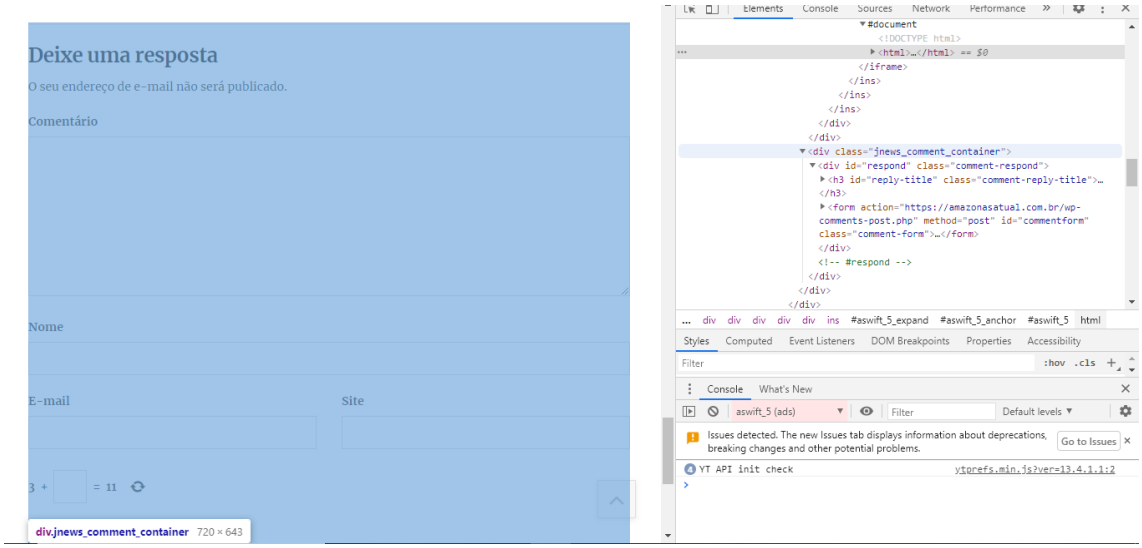
Figura 4 - Seção de Comentários - Exemplo 1



Fonte: Próprio autor.

Neste primeiro exemplo, retirado do site DevMedia, é possível perceber que a seção de comentários é identificada como “comentario”.

Figura 5 - Seção de Comentários - Exemplo 2



Fonte: Próprio autor.

Já no site Amazonas Atual, a seção de comentários é identificada como “respond”. Portanto, foi realizada uma pesquisa para identificar e conhecer as diversas expressões que representam a seção de comentários dos sites.

2.5 TÉCNICAS UTILIZADAS

2.5.1 Web Scraping, Web Crawler e Bots

Segundo Mitchell (2019), a extração de dados da Internet de forma automatizada surgiu quase ao mesmo tempo que o surgimento da Internet, ou seja, essa coleta de dados é antiga, porém era conhecida como *screen scraping*, *data mining, web harvesting*, entre outros. Hoje, o termo que prevalece é o Web Scraping.

Ainda de acordo com Mitchell (2019), Web Scraping é uma prática de coleta de dados através de um programa automatizado que consulta um servidor web, requisita dados, geralmente em HTML e outros arquivos que compõem as páginas web, e faz a manipulação de Strings dos dados para a extração de informações às quais se busca.

Já o conceito de Web Crawler “são programas que percorrem a Web, visitando um conjunto inicial de links, em busca de outros links na página de

maneira recursiva, até que uma condição de parada seja satisfeita“ (BORGES, 2018).

Neste projeto, não há a necessidade de percorrer toda a Web. E para isso, existem os chamados focused crawler ou crawlers focados. Os crawlers focados são programas configurados para percorrer um domínio específico, ou seja, apenas as páginas de interesse do usuário. (LIU, 2011 apud BORGES, 2018).

Os bots, também conhecidos como spider, são algoritmos desenvolvidos para a análise e extração de informações de páginas web de forma automatizada e sistemática (OMARI, 2016 apud GALDINO, 2020).

Sendo assim, os bots irão rastrear e raspar dados de sites, utilizando as técnicas de web crawler e web scraping de forma simultânea ou como tarefas distintas (KHALIL, 2017 apud GALDINO, 2020).

O sistema SEETER irá utilizar as técnicas de Web Scraping para coletar dados das páginas web cujos links foram cadastrados por usuários no sistema SEETER. É através desses recursos que será possível encontrar as seções de comentários dos sites e interpretá-las, identificando informações relevantes. A técnica também permite coletar o título da matéria para salvar no banco de dados do sistema e apresentar para o usuário. As técnicas serão utilizadas através da linguagem de programação Python e com a importação da biblioteca Beautiful Soup.

2.5.2 Scrum

De acordo com SCRUMstudy (2016, apud RAMOS, 2017), o Scrum é “um framework ágil para o gerenciamento de projetos complexos”. Ele visa se adaptar às demandas do projeto e tem como valores: a adaptabilidade, transparência, feedback e melhorias contínuas, dentre outros (SCRUMstudy, 2016 apud RAMOS 2017).

Segundo Sommerville (2011), o Scrum possui três fases principais, são elas: a fase de planejamento geral, na qual são estabelecidos os objetivos gerais do projeto e da arquitetura do software; a segunda fase são sequências de sprints e a cada uma delas é desenvolvido um incremento do sistema; e a terceira é a fase de encerramento do projeto, com elaboração de

documentações, manuais e avaliações de lições aprendidas ao decorrer do projeto.

Sendo assim, uma das principais características do Scrum são seus ciclos de sprint. Os ciclos geralmente duram duas a quatro semanas e neles são desenvolvidos uma parte do sistema (SOMMERVILLE, 2011).

Outras características também são importantes de serem destacadas, como: a elaboração do Sprint Planning, que é a fase onde o time irá definir o que será desenvolvido de acordo com o Product Backlog (RAMOS, 2017); o Product backlog é a lista do trabalho que deve ser desenvolvido, e onde também são identificadas as prioridades e riscos (SOMMERVILLE, 2011); a cada sprint são realizadas reuniões diárias com objetivo de planejar o que será desenvolvido diariamente e tornar visível o trabalho de cada membro (STREULE, 2016 apud RAMOS, 2017); no último dia de cada sprint também são realizadas a Sprint Review que é a revisão e adaptação do produto a ser entregue (ANTUNES, 2015 apud RAMOS, 2017) e Sprint Retrospective, onde é “revisado e melhorado o desempenho e maneira de trabalho do time Scrum” (SABBAGH, 2013 apud RAMOS 2017).

Para o desenvolvimento deste projeto serão utilizadas algumas características do Scrum: desenvolvimento em ciclos de Sprints e elaboração do Product Backlog.

As sprints terão 10 dias de duração, e serão divididas em 6. A cada uma delas, haverá um dia de revisão da sprint, realizada junto ao orientador do projeto. As tarefas a serem desenvolvidas também estão em um Product Backlog, divididas em 11 histórias do usuário e ordenadas de acordo com seus graus de prioridade.

2.6 TECNOLOGIAS UTILIZADAS

2.6.1 Python

Python é uma linguagem de programação gerida pela PSF (Python Software Foundation) e é conhecida por ser robusta e de fácil utilização.

“A linguagem oferece suporte a desktops, desenvolvimento web, aplicações mobile, geoprocessamento, processamento de imagens, robótica,

Data Science, programação para hardware (Harduíno e RaspbarryPi), desenvolvimento de games, biotecnologia e também no desenvolvimento científico, pois, trabalha com números grandes e complexos" (SILVA, 2019).

De acordo com Silva (2019), o Python é utilizado principalmente quando se tem como meta um desenvolvimento ágil e isso é possível por conta de a linguagem seguir a metodologia RAD (Rapid Application Development – Desenvolvimento rápido de aplicações), sendo assim, a linguagem evita desperdícios e diminui tempo e custos de desenvolvimento.

Portanto, como é uma linguagem com bastante vantagens e levando em conta que o conteúdo sobre Python é bastante amplo e diversificado na Internet, a linguagem foi escolhida para o desenvolvimento do projeto SEETER.

2.6.2 Django Rest Framework

Para o desenvolvimento do Back-End do Projeto SEETER foi escolhido o Django Rest Framework.

O Django Rest Framework foi desenvolvido com base no Django Framework visando auxiliar na implementação de APIs e na adequação a normas de segurança. O Django Rest Framework ainda fornece uma interface no navegador para que o usuário possa visualizar e interagir com a API desenvolvida (CHRISTIE, 2019, apud DE CASTRO, 2019). E o banco de dados utilizado foi o PostgreSQL, que está integrado com o Back-End.

2.6.3 Framework Angular

Já para a implementação do Front-End foi escolhido o Framework Angular. O Angular é um framework desenvolvido em JavaScript e é open source (código aberto). O Angular permite a utilização da sintaxe do HTML para a manipulação dos elementos que compõem a interface de maneira eficiente (BRANAS, 2014, apud BAGLIOTTI, 2020). Algumas das principais vantagens de se utilizar o Angular é de que ele é baseado em componentes, possui um melhor desempenho e possui um sistema de templates mais completo (SINGH, 2017, apud BAGLIOTTI, 2020).

3 METODOLOGIA

Este projeto foi desenvolvido de forma individual no período de aproximadamente seis meses. Portanto, foi escolhida a abordagem de desenvolvimento Incremental para guiar a implementação do sistema SEETER. Através do desenvolvimento incremental foi possível estruturar uma arquitetura com os principais componentes de um sistema para armazenar dados de sites de acesso público e seus respectivos comentários registrados, desenvolver aplicação com Back-end e Front-end integrados e um sistema que notifica os usuários sempre que a quantidade de comentários for alterada em um dos sites como é proposto nos objetivos gerais.

Esse modelo de processo de software consiste em:

> intercalar as atividades de especificação, desenvolvimento e validação. O sistema é desenvolvido como uma série de versões (incrementos), de maneira que cada versão adiciona funcionalidade à anterior (SOMMERVILLE, 2011, p. 20).

Adotar o modelo de desenvolvimento incremental trouxe vantagens à elaboração deste projeto, pois o modelo se adequa melhor às mudanças, a quantidade de análise e documentação a ser refeita é menor do que quando se opta pelo modelo em cascata. Outra vantagem é a de que é possível realizar entrega e implementação rápida de um software útil ao cliente, mesmo se todas as funcionalidades não forem incluídas e concluídas (SOMMERVILLE, 2011).

Na fase de Especificação foi realizada a análise e especificação dos requisitos do sistema, foram construídos o diagrama de Casos de Uso e a descrição de cada um dos casos, digrama de atividades, diagrama de classes, diagramas de sequência e mockups das telas do sistema.

Já na fase de Desenvolvimento foram desenvolvidas as arquiteturas do Software e do projeto, além dos módulos de Back-End, Frond-End e o Script de verificação dos sites. O sistema SEETER foi implementado utilizando o Django Rest Framework e a linguagem Python. Foi feita a implementação da seção do usuário, onde é apresentada a lista de links cadastrados por ele. Também foi desenvolvida a parte do sistema responsável por localizar a seção de comentários por meio das expressões regulares e o contador de comentários.

Além disso, foi feita a integração com o banco de dados e parte para notificar o usuário.

Na validação foram realizados principalmente os testes para verificar se as funcionalidades estão funcionando adequadamente, testes de integração com o banco de dados, testes de regressão a cada vez que uma nova versão foi implementada e testes de usabilidade para verificar a interface com o usuário. Também foram realizados alguns testes exploratórios para verificar se haviam erros em fluxos incorretos. E ao final do projeto, foi feita a verificação do sistema como um todo.

Portanto, o modelo de Desenvolvimento Incremental foi adotado, e especificação, desenvolvimento e validação foram atividades presentes durante todo o período de duração deste projeto.

4 PROJETO

4.1 ANÁLISE E ESPECIFICAÇÃO DE REQUISITOS

Para o desenvolvimento do sistema SEETER, primeiramente foi analisado como seria possível realizar a contagem dos comentários de um site. Nessa fase foi realizada a análise das expressões regulares e notou-se que era possível contar os comentários através do HTML de cada página web, utilizando o Web Scraping. Após isso, também foi decidido a criação de um banco de dados para armazenar dados de usuários e dos sites. E após isso, também foram criados os mockups para definir as principais funcionalidades de interação do sistema com o usuário.

Foi realizada uma análise a fim de conhecer as expressões regulares utilizadas para identificar as seções de comentários dos sites de acesso público e assim poder fazer o acompanhamento destas seções.

Para esta fase do projeto, apenas uma das expressões que representam a seção de comentários foi escolhida, a fim de demonstrar o correto funcionamento do sistema. A expressão regular escolhida se refere à seção de comentários do Blog Farol do Amazonas, que é um site de notícias sobre a Amazônia voltado para a área Educacional.

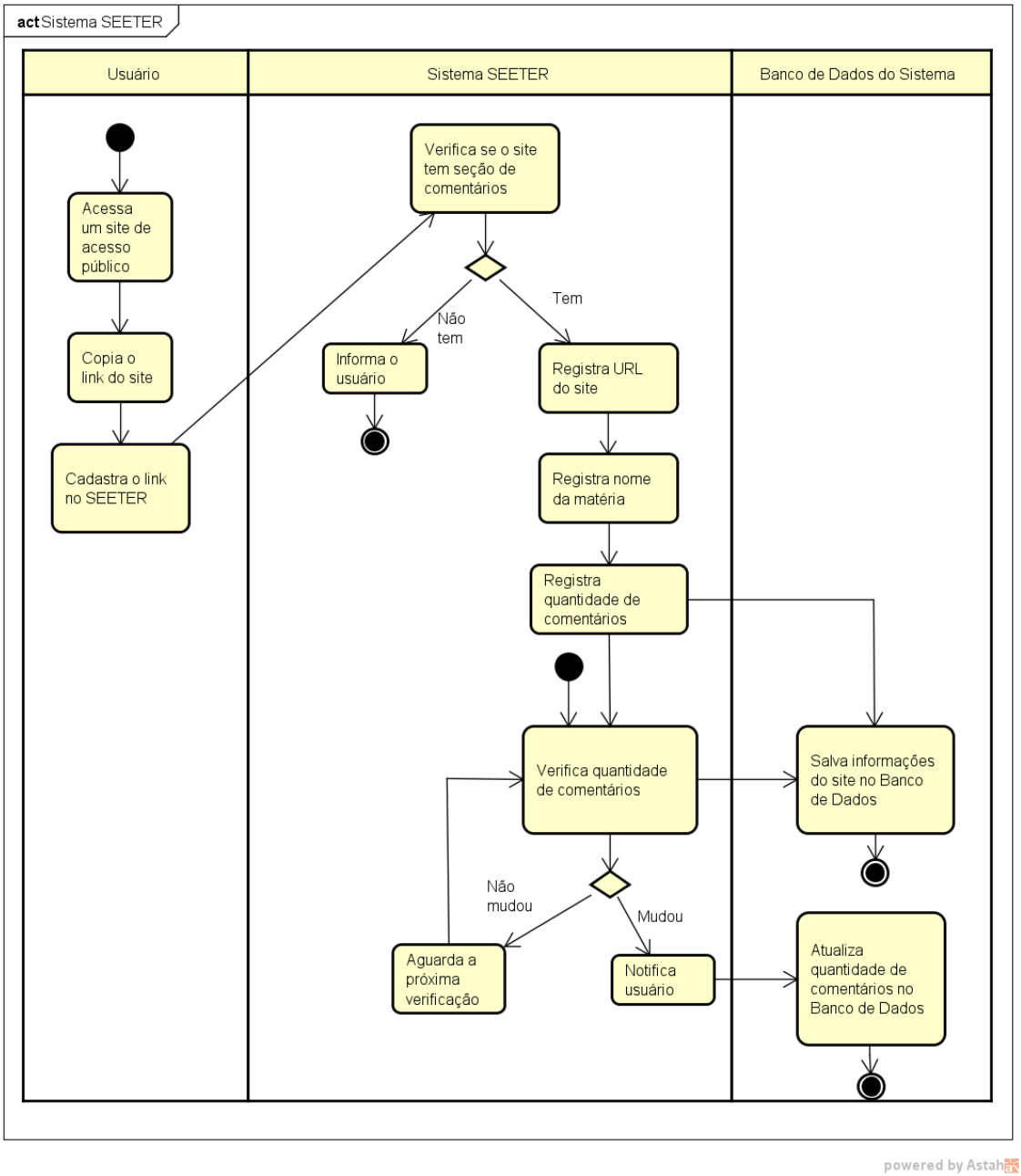
4.2 DIAGRAMA DE ATIVIDADES

O Diagrama de Atividades representa o fluxo das atividades realizadas por um sistema. O sistema SEETER tem início de suas atividades quando um usuário escolhe um site de acesso público que deseja acompanhar os comentários da página e copia o link. Então vai até o sistema SEETER, realiza seu login e cadastra o novo link.

O sistema, por sua vez, verifica se o site realmente possui uma seção de comentários. Caso o site não possua, o sistema notifica o usuário e encerra suas atividades. Caso encontre, o sistema irá registrar a URL do site, o nome da matéria (título da página), o nome do site, e a quantidade atual de comentários. Todas essas informações serão armazenadas no Banco de Dados do Sistema.

A cada 15 minutos, o sistema fará uma contagem na quantidade de comentários. Caso a quantidade não tenha mudado, o sistema apenas aguarda pela próxima contagem. Caso a quantidade tenha mudado, o SEETER notifica o usuário e atualiza a quantidade no Banco de Dados. Se houver alguma falha como uma queda de Internet, e o sistema precisar ser atualizado, ele retorna para onde o usuário estava a partir da parte de verificação da quantidade de comentários.

Figura 6 - Diagrama de Atividades



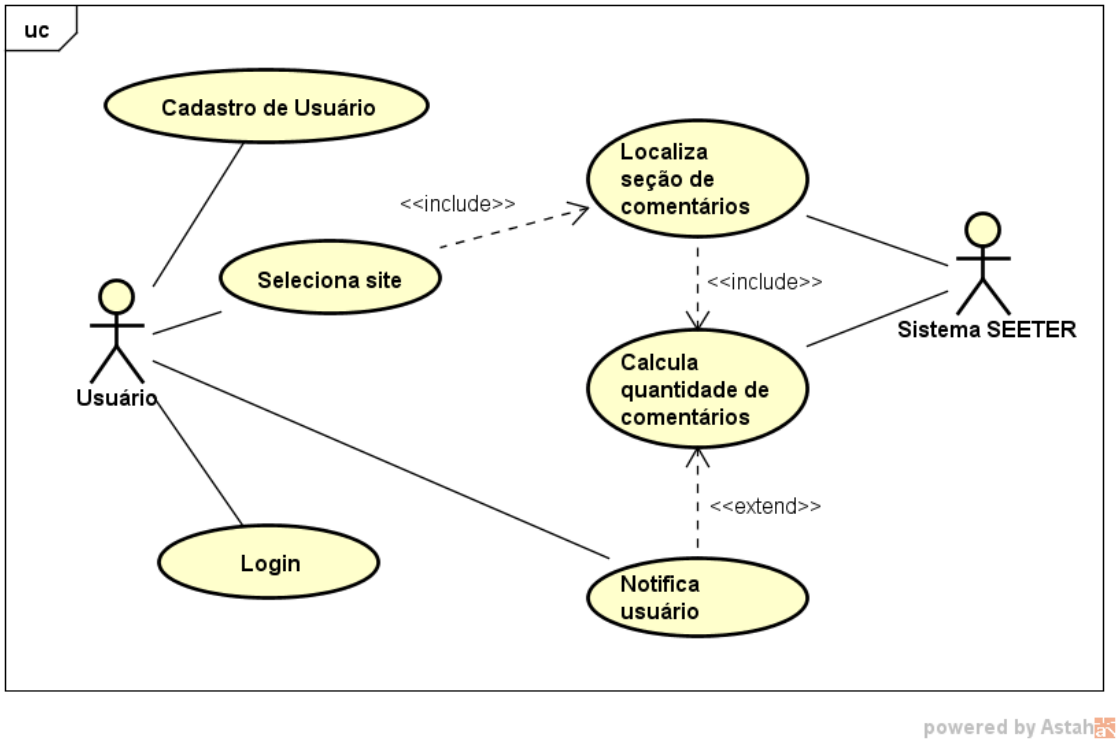
Fonte: Próprio autor.

4.3 DIAGRAMA DE CASOS DE USO

O Diagrama de Casos de Uso representa um fluxo de ações que serão realizadas pelo sistema a ser desenvolvido. O Diagrama de Casos de Uso abaixo mostra o fluxo principal do sistema e é representado pelos atores Usuário e Sistema SEETER.

Inicialmente o Usuário se cadastra no sistema ou realiza login e seleciona um site de acesso público no qual deseja acompanhar os comentários. O sistema SEETER localiza a seção de comentários e faz uma contagem na quantidade de comentários no momento que o usuário seleciona. O sistema irá recontar a quantidade de comentários constantemente e caso a quantidade tenha aumentado, ele notifica o usuário.

Figura 7 - Diagrama de Casos de Uso



Fonte: Próprio autor.

UC001:	Cadastrar Usuário
Objetivo:	Realizar cadastro do usuário no banco de dados do sistema.
Ator principal:	Usuários do SEETER

Ator secundário:	Sistema SEETER
Pré-condições:	Ainda não estar registrado no banco de dados do sistema.
Pós-condições:	
Cenário principal: O usuário irá até a seção de cadastro, adicionará todos os seus dados: nome de usuário, senha e e-mail; finalizará o cadastro.	
Ações do Ator	**Ações do sistema**
1 - Acessar a seção de cadastro.	
	2 - O sistema irá apresentar uma página para inserir os dados do cliente.
3 - Inserir os seus dados nos devidos campos.	
4 - Selecionar a opção " Cadastrar".	
	5 - O sistema realiza verificação para confirmar que o usuário ainda não está registrado no banco de dados.
	6 - O sistema adiciona o novo registro no banco de dados e exibirá mensagem de “Cadastro realizado com sucesso”.
	7 - O sistema irá finalizar o processo e exibirá a página principal do sistema para o usuário.
Cenário Alternativo 1: Usuário já registrado no banco de dados. O sistema exibe uma mensagem: “Usuário já cadastrado” e exibe a tela de Login.	
Cenário Alternativo 2: Usuário clica para se cadastrar deixando um ou mais campos em branco. O sistema exibe a mensagem “Por favor, preencha todos os campos obrigatórios” e destaca os campos que precisam ser preenchidos.	
Regras de negócio: O sistema só deve registrar usuários ainda não cadastrados. Todos os campos devem ser preenchidos obrigatoriamente (nome de usuário, login, senha, e-mail).	

UC002:	Selecionar site
Objetivo:	Usuário selecionar o site que deseja acompanhar os comentários e registrá-lo no sistema.
Ator principal:	Usuários do SEETER
Ator secundário:	Sistema SEETER
Pré-condições:	Usuário estar logado no sistema.
Pós-condições:	
Cenário principal: O usuário seleciona o site que deseja acompanhar os comentários, copia o link da página e registra no sistema SEETER.	
Ações do Ator	**Ações do sistema**
1 - Acessar a seção de cadastrar novo link.	
	2 - O sistema irá apresentar um card para inserir o link da página selecionada, como mostra a Figura 17.
3 - Inserir o link da página selecionada no devido campo.	
4 - Selecionar a opção " Adicionar".	
	5 - O sistema realiza verificação para confirmar que o link ainda não está registrado no banco de dados.
	6 - O sistema adiciona o novo registro no banco de dados e exibirá mensagem de “Cadastro realizado com sucesso”.
	7 - O sistema irá finalizar o processo e exibirá a listagem de links cadastrados pelo usuário.
Cenário Alternativo 1: Usuário insere link já registrado no banco de dados. O sistema exibe uma mensagem: “Link já cadastrado” e retorna à listagem de links.	
Cenário Alternativo 2: Usuário adiciona link de site com acesso restrito. Sistema informa ao usuário que o link não é válido para o sistema e retorna à listagem de links.	

Regras de negócio: O sistema deve registrar links que ainda não foram cadastrados pelo usuário no sistema e o link deve corresponder a uma página de acesso público.

UC003:	Localizar seção de comentários
Objetivo:	Sistema percorrer o site registrado pelo usuário e localizar a seção de comentários e contabilizar a quantidade de comentários no momento do registro.
Ator principal:	Sistema SEETER
Ator secundário:	
Pré-condições:	Site ser de acesso público.
Pós-condições:	
Cenário principal: Sistema percorre o site registrado pelo usuário até encontrar a seção de comentários, verificar a quantidade de comentários e registrar no sistema.	
Ações do Sistema	
1 - Acessar o link registrado pelo usuário.	
2 - Localizar a seção de comentários	
3 - Contabilizar a quantidade de comentários.	
4 - Registrar no Banco de Dados a quantidade atual de comentários.	
5 - Exibir na listagem de links a quantidade de comentários atual.	
Cenário Alternativo 1: Seção de comentários não encontrada. O sistema notifica o usuário que nenhuma seção de comentário foi encontrada e solicita novo link.	

UC004:	Calcular quantidade de comentários
Objetivo:	Contabilizar quantidade de comentários frequentemente e verificar se a quantidade foi notificada.
Ator principal:	Sistema SEETER
Ator secundário:	
Pré-condições:	Site possuir seção de comentários.
Pós-condições:	Notificar o usuário caso a quantidade

	mude.
Cenário principal: Frequentemente, o sistema irá contabilizar a quantidade de comentários no site registrado pelo usuário e comparar com a última quantidade registrada.	
Ações do Sistema	
1 - A cada 15 minutos o sistema acessa o site registrado pelo usuário.	
2 - Localiza a seção de comentários.	
3 - Contabiliza quantidade de comentários.	
4 - Verifica se a quantidade é diferente do último registro.	
5 - Atualiza o Banco de Dados	
Cenário Alternativo 1: O site bloqueia a seção de comentários. O sistema notifica o usuário do ocorrido.	
Cenário Alternativo 2: O site muda a URL e a página registrada no sistema não é encontrada. O sistema notifica o usuário e remove o link do Banco de Dados.	
Regras de negócio: A cada 15 minutos, verificar a quantidade de comentários nos sites registrados.	

UC005:	Notificar usuário
Objetivo:	Caso a quantidade de comentários em um dos sites registrados pelo usuário tenha mudado, enviar notificações pelo sistema SEETER.
Ator principal:	Sistema SEETER
Ator secundário:	
Pré-condições:	Quantidade de comentários ter sido atualizada.
Pós-condições:	

Cenário principal: Se a quantidade de comentários tiver mudado, o sistema irá atualizar o banco de dados e na página de listagem de links do usuário, irá notificar a nova quantidade de comentários registrada.	
Ações do Sistema	
1 - Atualizar a quantidade de comentários no Banco de Dados.	
2 - Atualizar a quantidade de comentários na listagem de links registrados pelo usuário (Figura 18).	
3 - Enviar mensagem de notificação pelo sistema SEETER.	

4.4 DIAGRAMA DE CLASSES

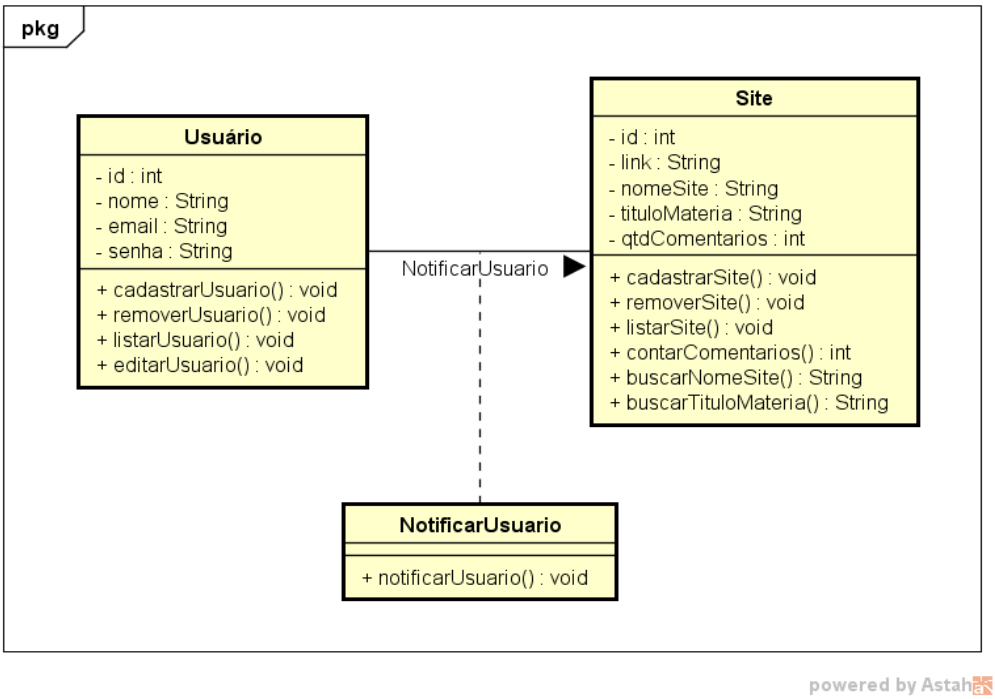
O Diagrama de Classes é uma representação das Classes de um sistema e seus respectivos atributos, métodos e a relação entre os objetos. O sistema SEETER terá duas classes principais: Usuário e Site; e uma classe de associação: NotificarUsuario.

A classe Usuário terá como atributos a identificação, o nome, e-mail e senha dos usuários. No banco de dados será possível realizar o cadastro, remoção, listagem e edição de dados dos usuários.

A classe Site terá os atributos de identificação, o link, nome do site, o título da matéria e a quantidade de comentários. No sistema será possível cadastrar, remover e listar os sites cadastrados. O sistema também fará uma busca a cada vez que o usuário cadastrar um novo link para buscar a quantidade de comentários, nome do site e título da matéria que está sendo cadastrado.

A classe NotificarUsuario é uma associação entre Usuário e Site. Ela terá um método para notificar usuários associados a ela, caso a quantidade de comentários mude.

Figura 8 - Diagrama de Classes



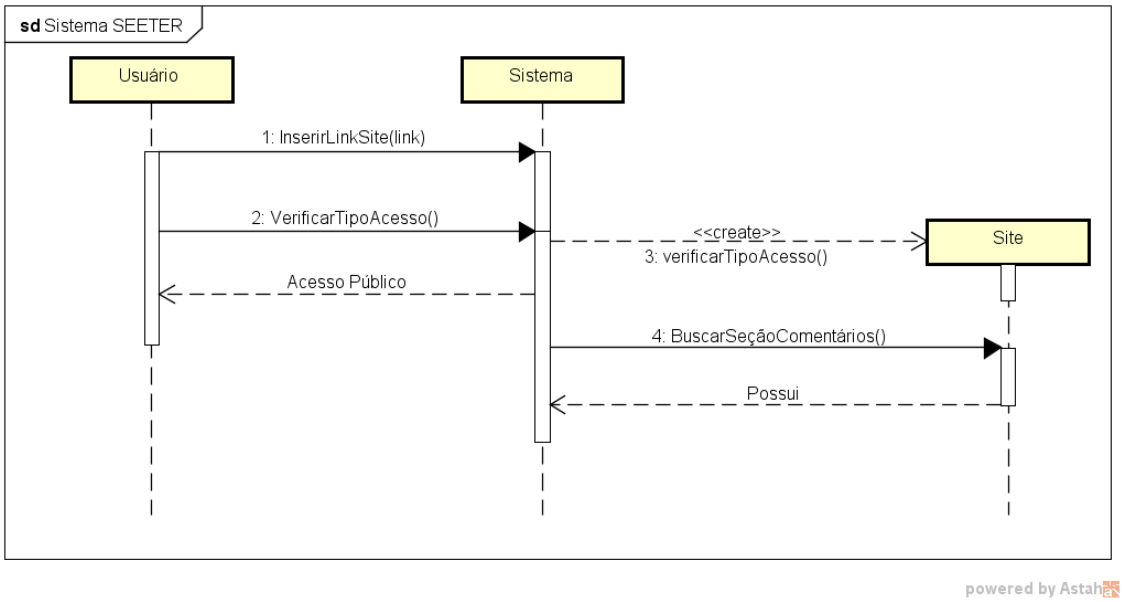
Fonte: Próprio autor

4.5 DIAGRAMA DE SEQUÊNCIA

O Diagrama de Sequência representa a interação do sistema, mostrando as mensagens que serão trocadas entre os objetos e classes do sistema e a ordem que elas acontecem. Os diagramas abaixo representam essas interações no sistema SEETER.

O primeiro diagrama representa o momento em que um usuário faz o cadastro de um novo link no sistema. O sistema verifica se o site tem acesso público, e caso tenha, retorna uma mensagem ao usuário que o cadastro do link foi realizado. Em seguida o sistema também procura se o site possui seção de comentários, caso exista, é retornada uma mensagem ao sistema.

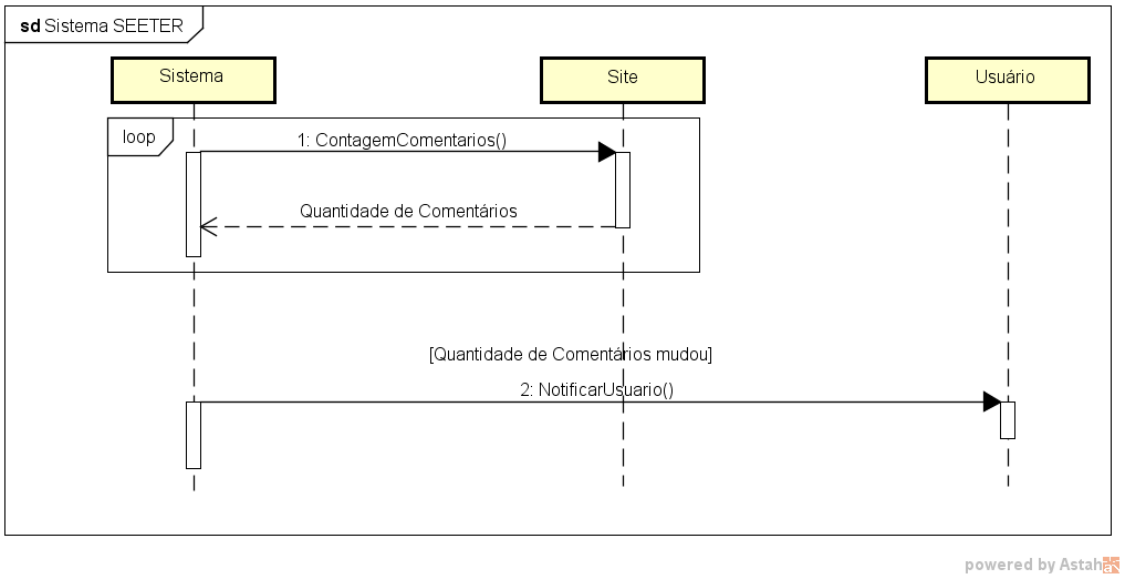
Figura 9 - Diagrama de Sequência 1



Fonte: Próprio autor

O segundo Diagrama de Sequência representa a contagem de comentários que ocorrerá a cada 15 minutos no sistema. O sistema verificará no site a quantidade de comentários que o site possui e irá comparar com a quantidade registrada no Banco de Dados, caso a quantidade de comentários tenha mudado, o sistema notifica o usuário.

Figura 10 - Diagrama de Sequência 2

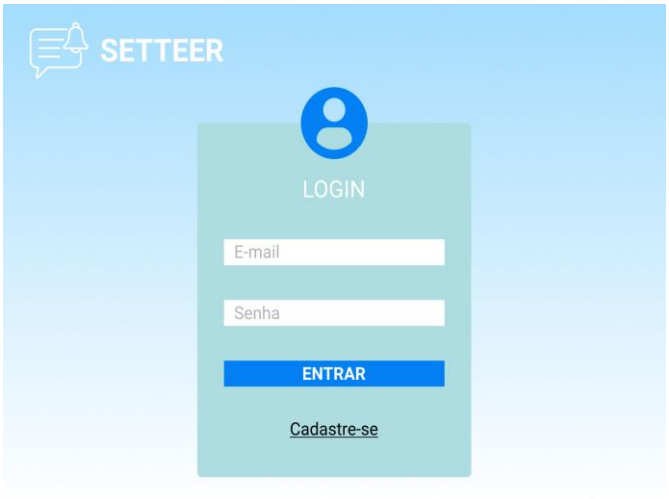


Fonte: Próprio autor

4.6 MOCKUPS DE TELAS DO SISTEMA

Os mockups de telas do Sistema auxiliaram a definir as telas que seriam utilizadas no sistema. Portanto, definiu-se que o sistema teria uma tela de login (Figura 11), uma tela principal (Figura 12), onde seriam listados os links cadastrados pelos usuários e onde teria um botão para adicionar um novo link. Quando houver uma atualização na quantidade de comentários a intenção é que seja mostrada uma notificação na lista de links, como mostra a Figura 13.

Figura 11 – Mockup da Tela de Login



Fonte: Próprio autor

Figura 12 – Mockup da Tela Principal do Sistema

Fonte: Próprio autor

Figura 13 – Mockup da Tela Principal do Sistema com notificação

Fonte: Próprio autor

5 FASE DE DESENVOLVIMENTO DO SISTEMA

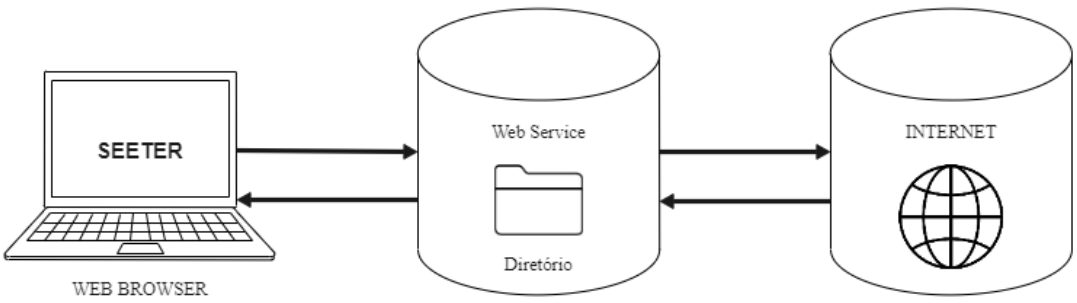
A fase de desenvolvimento do sistema é a parte responsável pela implementação da estrutura e dos códigos que formaram o sistema SEETER. Então, foi nessa fase que foi desenvolvida a arquitetura do software e do projeto, além dos módulos de Back-End, Front-End e Script responsáveis pelo funcionamento do sistema.

5.1 ARQUITETURA DE SOFTWARE

De acordo com Bass, L. *et al.* (2012 apud SOUZA, 2017) “a arquitetura de Software contém a descrição da estrutura do sistema, seus elementos e suas relações, que tem por objetivo expressar o comportamento do sistema.”

A Figura 14 representa a arquitetura do sistema SEETER. Primeiramente, o usuário irá cadastrar uma URL de um site que deseja acompanhar os comentários, então, no Web Service será realizada uma busca no diretório local para verificar se a URL já está presente no sistema. Caso a URL não esteja no diretório, o Web Service busca a URL na Internet. Nesse caso, a Internet alimentará o Web Service com a URL que o diretório ainda não possui e o Web Service fornecerá o serviço para o Web Browser.

Figura 14 - Arquitetura do Sistema SEETER



Fonte: Próprio autor

5.2 ARQUITETURA DO PROJETO

A Figura 15 representa a Arquitetura do Projeto SEETER. O Projeto é composto por 3 partes principais: Backend, responsável pela parte lógica do projeto, Front-end, responsável pela interface de interação com o usuário e um arquivo em Python.

O Backend é formado por 2 pacotes que são a Aplicação SEETER e o Controle SEETER e outros 3 arquivos: Manage.py e db.env e requirements.txt.

O pacote Aplicação SEETER possui pacotes e arquivos da parte da aplicação. O pacote API possui os arquivos serializers e viewsets. O pacote Migrations possui as classes do sistema, que são: Usuário, Site e Notificar Usuário. Além disso, o pacote possui os arquivos Python: admin, apps, models e tests, que são criados automaticamente ao se criar a aplicação.

O outro pacote é o de Controle do sistema, que ao ser criado também gera automaticamente os arquivos Python: asgi, settings, urls, wsgi e __init__.

O Front-end possui 2 arquivos que são criados automaticamente, que são o package.json e o package-lock.json e um pacote com todos os arquivos para configurar a interface e fazer a interação com o Backend.

Esse pacote responsável pela interface e integração é formado por dois outros pacotes, o node_modules e o src, além dos arquivos angular.json, package-lock.json, package.json, README.md, tsconfig.app.json e o tsconfig.json que são criados automaticamente.

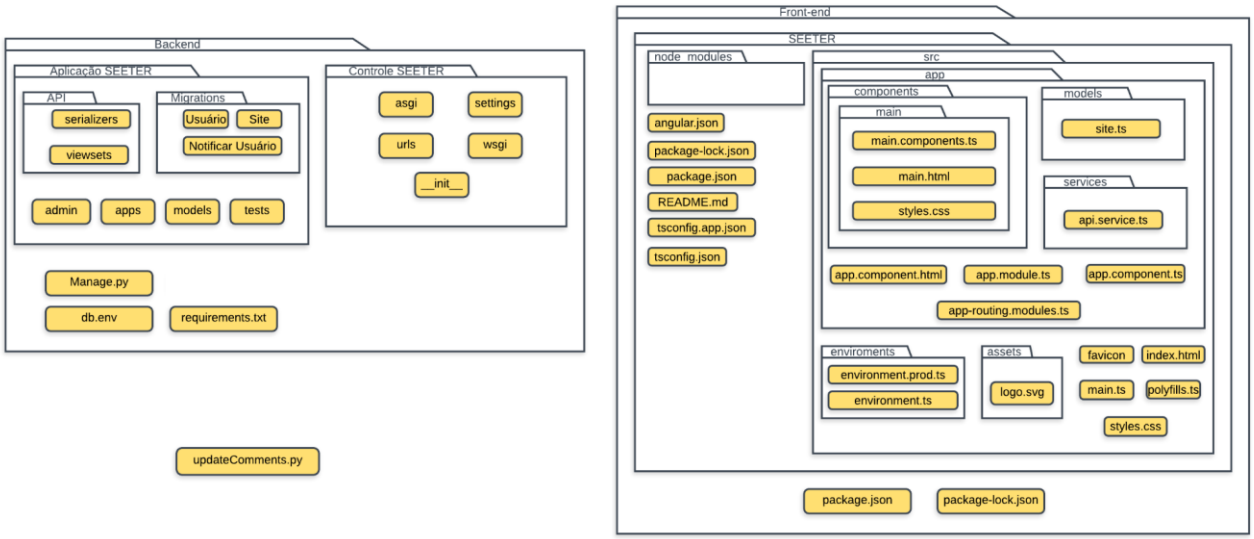
O pacote src, por sua vez, é composto de mais 3 pacotes e 5 arquivos. Os pacotes são o app, environments e assets. Os arquivos são favicon, index.html, main.ts, polyfills.ts e styles.css.

O app é formado pelos components, que é a pasta responsável pelos componentes da tela, e para cada componente são criados os arquivos main.components.ts, main.html e style.css. O models é formado pelos objetos que desejam ser apresentados. E o services tem o arquivo api.service.ts. Existem também os arquivos app.components.html, app.module.ts, app.component.ts e app-routing.modules.ts.

O pacote environments tem os arquivos environments.prod.ts e environment.ts.

O assets é composto por arquivos que se deseja separar, como, por exemplo, a logo.svg.

Figura 15 - Arquitetura do Projeto SEETER



Fonte: Próprio autor

5.3 MÓDULOS DO PROJETO

Os módulos do projeto SEETER foram desenvolvidos de acordo com os mockups produzidos na primeira fase deste Trabalho de Conclusão de Curso, seguindo a ideia de se ter uma listagem de sites e que essa listagem seria atualizada quando houvessem novos comentários. Porém, foram realizadas algumas adaptações na interface, por conta do Framework Angular que foi utilizado para o desenvolvimento do Front-end e o Framework Bootstrap que auxiliou na criação da interface a partir de componentes que já possuíam seus modelos.

Nesta primeira fase de desenvolvimento, o Projeto SEETER é composto por uma tela principal onde são listados os sites e um card para cadastro de novos sites.

A tela principal (Figura 16) é uma listagem dos sites cadastrados pelo usuário. A lista apresenta uma coluna com a numeração da linha, a segunda coluna é o título do site, a terceira é o título da Matéria, que é buscado por um web scraping, a quarta coluna é o link do site, a quinta é a quantidade de

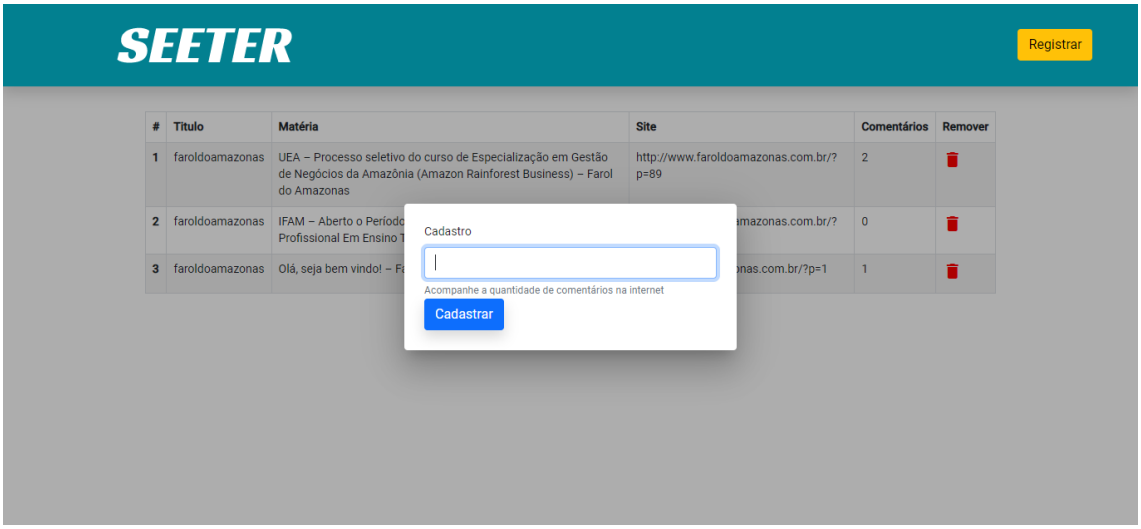
comentários, que também é coletada através de web scraping e a sexta coluna são ícones de lixeira, onde o usuário pode clicar e remover o site da listagem.

Figura 16 - Tela principal do SEETER

Fonte: Próprio autor

Além disso, na tela principal há um botão no canto superior direito, onde está escrito Registrar. Quando o usuário clicar no botão, o sistema abrirá um card (Figura 17) onde o usuário poderá adicionar o link que deseja acompanhar os comentários. Vale ressaltar que é necessário que o link adicionado possua “www.” após o “http://” ou “https://”, para que o sistema possa registrar corretamente o título do site.

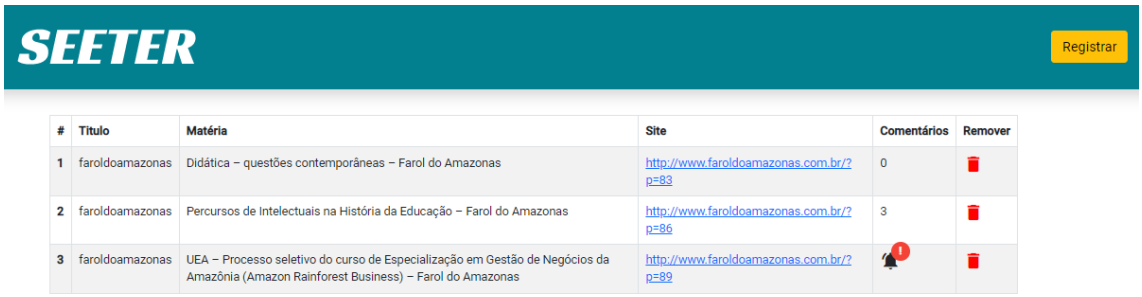
Figura 17 - Card de Cadastro de Sites



Fonte: Próprio autor

O sistema então, ficará acompanhando a quantidade de cada um dos sites a cada 15 minutos e assim que um dos sites modificar a quantidade de comentários, o sistema irá atualizar a lista de sites com a nova quantidade de comentários atualizada e um ícone de notificação ao lado do site onde houve a modificação (Figura 18).

Figura 18 – Notificação de novos comentários



Fonte: Próprio autor

6 RESULTADOS OBTIDOS

O Projeto SEETER teve como resultado um sistema capaz de armazenar links e realizar verificações constantes na quantidade de comentários dos sites registrados. Mais detalhadamente, os resultados obtidos foram:

- Banco de Dados capaz de armazenar os sites e usuários cadastrados no sistema e cumprindo o objetivo específico de implementar uma arquitetura para armazenar dados de sites de acesso público e seus respectivos comentários registrados.
- Back-End integrado ao banco de dados e com as lógicas da aplicação, além de conter a parte do Web Scraping.
- Front-End com as interfaces para os usuários, permitindo listar, cadastrar e remover os sites e suas principais informações. Também é onde o usuário receberá as notificações e verá as quantidades dos comentários. Além disso, está integrado ao Back-End, alcançando assim, outro objetivo específico que é o de desenvolver uma aplicação com Back-end e Front-end integrados de forma que permita conexão com banco de dados, lógica do negócio e interface para os usuários.
- Web Scraping para coletar o título de matérias e a quantidade de comentários através de links registrados pelos usuários.
- Pesquisa sobre as expressões regulares que representam a seção de comentários e identificação de uma das expressões para compor o Web Scraping do sistema.
- Script capaz de realizar verificação constante na quantidade de comentários nos sites registrados, utilizando Web Scraping. E com esses resultados, foi possível concluir o terceiro objetivo específico que é o de desenvolver o Sistema SEETER sendo capaz de notificar o usuário sempre que a quantidade de comentários for atualizada em sites registrados.

7 CONCLUSÕES DA PESQUISA E TRABALHOS FUTUROS

O projeto SEETER foi um projeto bastante desafiador. Nele foi necessário aprofundar o conhecimento nos frameworks Django-Rest Framework e Angular, para desenvolver utilizando tecnologias que já possuem recursos que auxiliam o desenvolvedor na hora de implementar o software.

Além disso, também foi necessário aprofundar as pesquisas em Web Scraping, para que fosse possível extrair as informações corretas dos sites através do link informado pelos usuários.

Neste primeiro momento, o projeto SEETER foi desenvolvido para atender o site Farol do Amazonas, a fim de demonstrar que é possível acompanhar os comentários de um site através do Web Scraping.

Para trabalhos futuros, pretende-se finalizar a sessão dos usuários, adicionando as telas de login e qualquer outra que seja observada a necessidade ao longo da utilização, como por exemplo, uma tela de edição das informações dos usuários ou dos sites cadastrados.

Pretende-se ainda, adaptar o sistema SEETER, para que ele atenda um número maior de sites, como por exemplo, conseguir identificar a seção de comentários de blogs e sites de notícias. E implementar a notificação via e-mail, para que a cada novo comentário, uma mensagem seja enviada por e-mail ao usuário, caso ele tenha habilitado essa opção de notificação.

Também ficará como um trabalho futuro transformar o SEETER em um Plug-in, que ficará presente em sites de acesso público, assim como existem atualmente os plug-ins do WhatsApp e do Facebook, para compartilhamento de notícias. Assim também é esperado que seja o plug-in do SEETER, onde será possível que os usuários cliquem e registrem automaticamente o link do site acessado no sistema.

Portanto, o projeto SEETER foi bastante enriquecedor aos conhecimentos e está de fato cumprindo sua missão de permitir que usuários acompanhem páginas de Sites de Acessos públicos e notificá-los sempre que houverem mudanças na quantidade de comentários.

REFERÊNCIAS

BAGLIOTTI, I. R.; GIBERTONI, D. **Reusabilidade no Desenvolvimento de um Sistema Web Utilizando o Framework Angular**. Revista Interface Tecnológica, v. 17, n. 1, 2020. DOI: 10.31510/infa.v17i1.826. Disponível em: https://revista.fatectq.edu.br/index.php/interfacetecnologica/article/view/826. Acesso em: 10 ago. 2021.

BORGES, T. C.; GANIMI, Z. O. **Extração de Dados com Web Scraping para Análise da Variação de Preço de Veículos Automotores**. 2018. 53 f. Trabalho de Conclusão de Curso (Graduação em Tecnologia em Sistemas de Computação) - Universidade Federal Fluminense, Niterói.

CGI - Comitê Gestor da Internet no Brasil. **Pesquisa sobre o uso das tecnologias de informação e comunicação nos domicílios brasileiros**: TIC domicílios 2018. São Paulo: CGI, 2019. 392 p. Disponível em: https://www.cetic.br/media/docs/publicacoes/2/12225320191028-tic_dom_2018_livro_eletronico.pdf. Acesso em: 19 mar. 2020.

COELHO, T. 10 fatos importantes sobre o uso de Internet no Brasil. **TechTudo.** [S.l.], 26 fev. 2018. Disponível em: https://www.techtudo.com.br/noticias/2018/02/10-fatos-importantes-sobre-o-uso-de-internet-no-brasil.ghtml. Acesso em: 19 mar. 2020.

CRISTOVÃO, H. M. **Um modelo híbrido de Recuperação de Informação e Conhecimento baseado na síntese de Mapas Conceituais obtidos por operações de Transformação de redes complexas orientadas por busca de relacionamentos entre Termos de Consulta em Bases de Dados ligados.** 2016. 315 p. Tese (Doutorado em Ciência da Informação) - Faculdade de Ciência da Informação, Universidade de Brasília, Brasília, 2016.

DAOUD, C. M. **Processamento eficiente de consultas em Sistemas de Busca.** 2016. 68 p. Tese (Doutorado em Informática) - Instituto de Computação, Universidade Federal do Amazonas, Manaus, 2016.

DE CASTRO, B. M. N; OLIVEIRA, F. H. M. Análise exploratória de dados geográficos de componente nativo para aplicativos híbridos. *In*: ESCOLA REGIONAL DE INFORMÁTICA DE GOIÁS (ERI-GO), 7., 2019, Goiânia. **Anais** [...]. Porto Alegre: Sociedade Brasileira de Computação, 2019.

GAIOSO, R. R. A. **Paralelização de algoritmos de busca de documentos mais relevantes na Web utilizando GPUS.** 2019. 132 p. Tese (Doutorado em

Ciência da Computação) - Centro de Ciências Exatas e de Tecnologia, Universidade Federal de São Carlos, São Carlos, 2019.

GALDINO, I. M.; GALLINDO, E. L.; MOREIRA, M. W. L. Utilização de Bots para Obtenção Automática de Dados Públicos usando as Técnicas de Web Crawling e Web Scraping. *In*: WORKSHOP DE COMPUTAÇÃO APLICADA EM GOVERNO ELETRÔNICO (WCGE), 8., 2020, Cuiabá. **Anais** [...]. Porto Alegre: Sociedade Brasileira de Computação, 2020. p. 172-179. DOI: https://doi.org/10.5753/wcge.2020.11269. Disponível em: https://sol.sbc.org.br/index.php/wcge/article/view/11269/11132. Acesso em: 1 mai. 2021.

INÁCIO, L. G. V. S. **Classificação de documentos jurídicos através de reconhecimento óptico de caracteres e expressões regulares: estudo de caso em uma empresa prestadora de serviços com o uso de uma ferramenta RPA.** 2020. 23 p. Trabalho de Conclusão de Curso (Graduação em Tecnologia em Análise e Desenvolvimento de Sistemas) - Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – Campus Hortolândia, São Paulo.

IWS - Internet World Stats. **Internet Usage Statistics**. [S. l.]: IWS, 2020. Disponível em: https://www.internetworldstats.com/stats.htm. Acesso em: 19 mar. 2020.

LAVADO, T. Uso da internet no Brasil cresce, e 70% da população está conectada. **G1,** [S. l.], 20 ago. 2019. Disponível em: https://g1.globo.com/economia/tecnologia/noticia/2019/08/28/uso-da-internet-no-brasil-cresce-e-70percent-da-populacao-esta-conectada.ghtml. Acesso em: 19 mar. 2020.

MITCHELL, R. **Web Scraping com Python**. 2. ed. São Paulo: Novatec, 2019.

NETCRAFT. **April 2019 Web Server Survey.** [S. l.], 2019. Disponível em: https://news.netcraft.com/archives/2019/04/22/april-2019-web-server-survey.html. Acesso em: 23 mar. 2020.

NOGUEIRA, L. Dados mostram o crescimento impressionante da internet em 10 anos. **Olhar Digital.** [S. l.], 17 mai. 2019. Disponível em: https://olhardigital.com.br/noticia/dados-mostram-o-crescimento-impressionante-da-internet-em-10-anos/85914. Acesso em: 19 mar. 2020.

PATRICIO, T. S.; MAGNONI, M. DA G. Mineração de Dados e Big Data na Educação. **Revista GEMInIS**, São Paulo, v. 9, n. 1, p. 57-75, 22 jun. 2018.

RAMOS, A. B.; JUNIOR, D. C. V. A Influência do Papel do Scrum Master no Desenvolvimento de Projetos Scrum. **Revista de Gestão e Projetos**, v. 8, n. 3, 2017. Disponível em: https://periodicos.uninove.br/gep/article/view/9677/4422. Acesso em: 1 mai. 2021.

SILVA, I. R. S.; SILVA, R. O. Linguagem de Programação Python. **Revista Tecnologias em Projeção**, v. 10, n. 1, 2019. Disponível em: http://revista.faculdadeprojecao.edu.br/index.php/Projecao4/article/view/1359/1064. Acesso em: 9 ago. 2021.

SOBRAL, N. V. *et al.* Estratégia para a recuperação de informação científica sobre as doenças tropicais negligenciadas: análise comparativa da Scopus, Pubmed e Web of Science. **Revista Cubana de Información en Ciencias de la Salud,** [S.l.], 2018.

SOMMERVILLE, Ian. **Engenharia de Software.** Tradução: Ivan Bosnic e Kalinka G. de O. Gonçalves. 9. ed. São Paulo: Pearson Prentice Hall, 2011. 529 p.

SOUZA, N. M. *et al.* **Relação entre Arquitetura de Software e Teste de Software: um Mapeamento Sistemático.** 2017. Relatório Técnico - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2017.