



UFOP

Universidade Federal
de Ouro Preto

**Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Departamento de Computação e Sistemas**

**Desenvolvimento de uma plataforma
web para monitoramento das mídias
sociais das empresas juniores da
Universidade Federal de Ouro Preto**

Sander Henrique Santos

**TRABALHO DE
CONCLUSÃO DE CURSO**

ORIENTAÇÃO:

Fernando Bernardes de Oliveira

COORIENTAÇÃO:

Filipe Nunes Ribeiro

**Dezembro, 2019
João Monlevade–MG**

Sander Henrique Santos

Desenvolvimento de uma plataforma *web* para monitoramento das mídias sociais das empresas juniores da Universidade Federal de Ouro Preto

Orientador: Fernando Bernardes de Oliveira

Coorientador: Filipe Nunes Ribeiro

Monografia apresentada ao curso de Sistemas de Informação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

Universidade Federal de Ouro Preto

João Monlevade

Dezembro de 2019

SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

S237d Santos, Sander Henrique .

Desenvolvimento de uma plataforma web para monitoramento das mídias sociais das empresas juniores da Universidade Federal de Ouro Preto. [manuscrito] / Sander Henrique Santos. - 2019.

56 f.

Orientador: Prof. Dr. Fernando Bernardes de Oliveira.

Coorientador: Prof. Dr. Filipe Nunes Ribeiro.

Monografia (Bacharelado). Universidade Federal de Ouro Preto. Instituto de Ciências Exatas e Aplicadas. Graduação em Sistemas de Informação .

1. Redes sociais on-line. 2. Aplicações Web - Negócios. 3. Processo decisório. 4. Inteligência competitiva (Administração). 5. Software de aplicação - Desenvolvimento. I. Oliveira, Fernando Bernardes de. II. Ribeiro, Filipe Nunes. III. Universidade Federal de Ouro Preto. IV. Título.

CDU 004:658

Bibliotecário(a) Responsável: Flavia Reis - CRB6 - 2431



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
INSTITUTO DE CIÊNCIAS EXATAS E APLICADAS
DEPARTAMENTO DE COMPUTAÇÃO E SISTEMAS

**FOLHA DE APROVAÇÃO****Sander Henrique Santos****Desenvolvimento de uma plataforma web para o monitoramento
das redes sociais das empresas juniores da Universidade Federal de Ouro Preto**

Membros da banca

Fernando Bernardes de Oliveira - Doutor/Orientador - Decsi/Ufop
Filipe Nunes Ribeiro - Doutor/Coorientador - Decsi/Ufop
Lucinéia Souza Maia - Doutora/Examinadora - Decsi/Ufop
Gilda Aparecida de Assis - Doutora/Examinadora - Decsi/Ufop

Versão final

Aprovado em 13 de dezembro de 2019

De acordo

Fernando Bernardes de Oliveira



Documento assinado eletronicamente por **Fernando Bernardes de Oliveira, PROFESSOR DE MAGISTERIO SUPERIOR**, em 04/10/2020, às 16:20, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0089527** e o código CRC **D0907029**.

Referência: Caso responda este documento, indicar expressamente o Processo nº 23109.007307/2020-81

SEI nº 0089527

R. Diogo de Vasconcelos, 122, - Bairro Pilar Ouro Preto/MG, CEP 35400-000
Telefone: - www.ufop.br

*Este trabalho é dedicado à minha mãe, que sempre esteve comigo em todos os momentos,
me apoiou e me deu forças para continuar.*

Agradecimentos

Agradeço aos meus pais, em especial minha mãe por não medir esforços para minha felicidade, pelo apoio e amor incondicional.

Agradeço a todos que passaram pelo meu caminho ao longo dessa jornada, vocês fizeram toda a diferença na minha vida e me mostraram como estar cercado de pessoas especiais faz a vida valer a pena.

Agradeço do meu coração ao meu orientador Fernando Bernardes por todo suporte, e por estar sempre preocupado com a saúde mental de seus alunos, o mundo precisa de mais profissionais como você. Ao meu coorientador Filipe Ribeiro o meu muito obrigado pela ajuda. Vocês foram fundamentais para o meu crescimento!

Ao Movimento Empresa Júnior, gratidão eterna por toda experiência! Em especial à Visão Tecnologias e Sistemas, a empresa júnior que se tornou minha família, que foi essencial para o meu crescimento pessoal e profissional. Ao Core - Núcleo de Empresas Juniores da UFOP, meu muito obrigado por tudo, em especial à Diretoria Executiva 2019.

Muito obrigado ao Sander do passado e do presente, sem sua força de vontade e sua busca constante por transformação nada disso teria se tornado realidade.

“Estamos aqui para fazer alguma diferença no universo, se não, por que estar aqui?”

— Steve Jobs

Resumo

Há uma grande quantidade de dados nas redes sociais que podem perder seu valor em meio a tanta informação. A análise e a mensuração desses dados podem auxiliar nas decisões estratégicas das organizações. Entre as organizações estão as empresas juniores, que utilizam das redes sociais de maneira estratégica. As redes sociais utilizadas são gratuitas e podem trazer um retorno positivo para as organizações. O trabalho proposto tem como objetivo desenvolver uma plataforma web para monitoramento das empresas juniores da Universidade Federal de Ouro Preto, baseada nas suas respectivas páginas do *Facebook*. O propósito desta plataforma é auxiliar a visualização e interpretação das principais informações sobre público de tais empresas, bem como as métricas que indicam o seu crescimento e, por fim, colaborar nas decisões estratégicas da organização. Os dados analisados na plataforma são extraídos utilizando a [API](#) do *Facebook* intitulada *Graph API*. Os dados passam por um processo de tratamento permitindo recuperar informações importantes. A aplicação proposta apresenta os resultados para os usuários de maneira prática e funcional, facilitando a visualização e análise, e contribuindo para tomada de decisão na organização.

Palavras-chaves: Redes sociais. Métricas. Sistema *Web*. Visualização. Tomada de decisão.

Abstract

There is a lot of data on social networks that can lose its value amid so much information. The analysis and measurement of this data can assist in the strategic decisions of organizations. Amongst organizations there are junior enterprises that use social networking strategically. The proposed work aims to develop a web platform for monitoring junior enterprises at the Federal University of Ouro Preto, based on their respective Facebook pages. The purpose of this platform is to assist in the visualization and interpretation of the main public information of these enterprises, as well as the metrics that indicate their growth, and collaborate in the strategic decisions of the organization. The data analyzed on the platform is extracted using the Facebook API entitled Graph API. The data goes through a treatment process and can retrieve important information. The proposed application presents the results to users in a practical and functional manner, facilitating visualization and analysis, contributing to decision making in the organization.

Key-words: Social networks. Metrics. Web system. Visualization. Decision making.

Lista de ilustrações

Figura 1 – Processo de KDD	18
Figura 2 – <i>Facebook Insights</i> - Página inicial	22
Figura 3 – <i>mLabs</i> - Página inicial	23
Figura 4 – Fluxo <i>OAuth 2.0</i> para obtenção das chaves e <i>tokens</i> de acesso a <i>Graph API</i>	26
Figura 5 – Transição das informações	28
Figura 6 – Diagrama Lógico - Relatórios da <i>fan page</i>	29
Figura 7 – Diagrama Lógico - Relatórios de Postagens	30
Figura 8 – Exemplo de <i>migration</i> da tabela impressões páginas	32
Figura 9 – Criação de uma aplicação a partir do <i>Facebook Developers</i>	33
Figura 10 – Seleção de permissões para geração do <i>token</i> do usuário	34
Figura 11 – Obtenção do <i>token</i> da <i>fan page</i>	34
Figura 12 – Obtenção da chave secreta e ID do aplicativo	35
Figura 13 – <i>Tokens</i> e chave secreta armazenadas no formato JSON	35
Figura 14 – Código de autenticação da <i>Graph API</i>	36
Figura 15 – Função de relatório de página	36
Figura 16 – Recuperação de curtidas da <i>fan page</i>	37
Figura 17 – Recuperação de publicações	37
Figura 18 – Execução do <i>script</i>	38
Figura 19 – Tela de <i>login</i>	39
Figura 20 – Página inicial da plataforma	40
Figura 21 – Tratamento dos dados e construção dos gráficos no <i>Controller</i>	40
Figura 22 – Evolução anual da métrica <i>page_fans</i>	41
Figura 23 – Categorização das impressões por faixa etária e gênero	42
Figura 24 – Relatório por período da <i>fan page</i>	43
Figura 25 – Evolução anual da métrica <i>post_engaged_fan</i>	44
Figura 26 – Exibição da postagem e suas métricas	45
Figura 27 – Relatório por período das publicações da <i>fan page</i>	46
Figura 28 – Diagrama completo do Banco de Dados	52

Lista de tabelas

Tabela 1 – Métricas da <i>fan page</i>	27
Tabela 2 – Parâmetros das publicações	28
Tabela 3 – Métricas das publicações	28

Lista de abreviaturas e siglas

API *Application Programming Interface*

CSS *Cascading Style Sheets*

HTML *Hypertext Markup Language*

HTTP *Hypertext Transfer Protocol*

IBGE Instituto Brasileiro de Geografia e Estatística

JSON *JavaScript Object Notation*

KDD *Knowledge Discovery in Databases*

MIT *Massachusetts Institute of Technology*

MVC *Model–View–Controller*

PDF *Portable Document Format*

PHP *PHP: Hypertext Preprocessor*

SQL *Structured Query Language*

URL *Uniform Resource Locator*

Sumário

1	INTRODUÇÃO	14
1.1	O problema de pesquisa	15
1.2	Objetivos	16
1.3	Metodologia	16
1.4	Organização do trabalho	17
2	REVISÃO BIBLIOGRÁFICA	18
2.1	Mineração de dados em Redes Sociais	18
2.2	Framework	19
2.3	<i>Laravel Charts</i>	20
2.4	Banco de Dados	20
2.5	Python	20
2.6	Graph API	20
2.7	Trabalhos correlatos	21
2.7.1	Facebook Insights	21
2.7.2	<i>mLabs</i>	22
2.7.3	Comparações com a aplicação proposta	23
2.8	Considerações finais	24
3	MODELAGEM DA APLICAÇÃO	25
3.1	Definição do escopo e do problema	25
3.2	Estudo de ferramentas e tecnologias	25
3.3	Coleta de dados no <i>Facebook</i>	26
3.3.1	Métricas	27
3.4	Banco de dados	29
3.5	Considerações finais	30
4	IMPLEMENTAÇÃO DA APLICAÇÃO	31
4.1	Desenvolvimento do sistema	31
4.1.1	Banco de dados no <i>Laravel</i>	31
4.1.2	Coleta dos dados a partir da <i>Graph API</i>	32
4.1.3	Autenticação dos usuários na plataforma	38
4.1.4	Página inicial da plataforma	39
4.1.5	Relatório anual da <i>fan page</i>	41
4.1.6	Relatório por período da <i>fan page</i>	42
4.1.7	Relatório das publicações da <i>fan page</i>	43

4.1.8	Relatório por período das publicações da <i>fan page</i>	44
4.2	Avaliações da plataforma	46
4.3	Considerações finais	47
5	CONCLUSÃO	48
	Conclusão	48
5.1	Trabalhos futuros	48
	REFERÊNCIAS	49
	APÊNDICES	51
	APÊNDICE A – DIAGRAMA DO BANCO DE DADOS	52

1 Introdução

As redes sociais estão cada vez mais presentes na vida das pessoas. Segundo um estudo feito pelo Instituto Brasileiro de Geografia e Estatística (IBGE) referente a uma amostragem até o final de 2016, o Brasil já tem 116 milhões de usuários de *internet*. Desse resultado, 94,2% dos brasileiros já enviaram ou receberam mensagens de texto, voz ou imagens, o que inclui redes sociais como o *Facebook* (IBGE, 2018). De acordo com Ferreira (2011), as redes sociais são caracterizadas por um conjunto de pessoas que possuem um padrão de contatos ou interação, e estabelecem diversos tipos de relações e, por meio delas, circulam diversos fluxos de informação.

Segundo um estudo realizado pelo portal Statista, o *Facebook* é uma das redes sociais mais utilizadas no mundo com mais de 2 bilhões de usuários (STATISTA, 2019). Criado em 2004, o *Facebook* tem como objetivo auxiliar as pessoas a criarem comunidades e aproximá-las. Os usuários usam a plataforma para manter contato com seu ciclo social, descobrir o que está acontecendo no mundo, além de compartilhar e expressar o que é importante para elas (FACEBOOK, 2019c).

Várias empresas já utilizam o *Facebook* como maneira de se aproximarem dos seus públicos, já que há uma maior interação entre a organização e os usuários. Geralmente, as organizações possuem *fan page*, que significa “página de fãs”, na qual são realizadas publicações que possam interessar aos públicos. Desse modo, a empresa consegue atrair diretamente seu público alvo, e conseqüentemente consegue mais seguidores. A página gera dados importantes para a organização, com os quais é possível conhecer o seu público, quantas pessoas curtiram, comentaram e compartilharam sua publicação, e conseqüentemente é uma maneira de mensurar os resultados obtidos pela estratégia aplicada na página (AMARAL et al., 2015).

Entre as organizações que utilizam o *Facebook* como ferramenta de mensuração de resultados, estão as empresas juniores que são uma “associação civil gerida por estudantes matriculados em cursos de graduação de instituições de ensino superior, com o propósito de realizar projetos e serviços que contribuam para o desenvolvimento acadêmico e profissional dos associados, capacitando-os para o mercado de trabalho” (BRASIL, 2016). As empresas juniores fazem parte do Movimento Empresa Júnior, que surgiu na França em 1967 e chegou ao Brasil em 1988. Atualmente, existem mais de 600 empresas juniores espalhadas pelo país compostas por alunos de diversas universidades, que realizam projetos e serviços que contribuem para o crescimento acadêmico e profissional (JUNIORES, 2019). No dia 6 de Abril de 2016 foi aprovada a Lei N° 13.267 que regulamenta as empresas juniores no Brasil (BRASIL, 2016). A Universidade Federal de Ouro Preto possui o Core – Núcleo de

Empresas Juniores ¹, responsável por acompanhar e dar suporte as 25 empresas juniores vinculadas a ele. O propósito do movimento é a busca por um Brasil Empreendedor, com pessoas comprometidas e capazes de transformar o Brasil por meio da realização de mais e melhores projetos (JUNIORES, 2019).

Atualmente, as empresas utilizam do *Facebook* como ferramenta de prospecção passiva, criando conteúdos relevantes para seus públicos. Grande parte das empresas juniores utilizam essa estratégia por ser gratuita e trazer retorno. Entretanto, após definidas as estratégias que serão utilizadas no *Facebook* para atrair e fidelizar o público, é necessário visualizar os resultados. As páginas contam com o *Facebook Insights*, que traz informações sobre as publicações, engajamento da página e seu público alvo. Porém, a interpretação pode se tornar confusa para quem não possui conhecimento prévio referente ao assunto.

A proposta deste trabalho é o desenvolvimento de uma plataforma *web* intitulada *Rocket*, baseada nos dados fornecidos pelas páginas das empresas juniores. O seu principal objetivo é facilitar a visualização dos dados e extrair as informações de maneira sucinta para aperfeiçoar a tomada de decisão.

1.1 O problema de pesquisa

De acordo com Santos (2013), a *web* possui o maior repositório de informações existentes no mundo. As pessoas estão em constante interação compartilhando um grande volume de dados que podem perder seu valor em meio a tanta informação. Em detrimento dessa circunstância, as organizações acabam tendo como maior dificuldade a mensuração e análise dos dados essenciais para auxiliar nas suas decisões estratégicas. Rezende et al. (2003) afirma que “devido à incapacidade do ser humano de interpretar tamanha quantidade de dados, muita informação e conhecimento, possivelmente úteis, podem estar sendo desperdiçados, ficando ocultos dentro das bases de dados espalhadas pelo mundo”.

As redes sociais geram uma grande quantidade de dados, que podem ser analisadas na perspectiva da Inteligência de Negócios, ou *Business Intelligence*. De acordo com Evelson (2008), a Inteligência de Negócios é “um conjunto de metodologias, processos, arquiteturas e tecnologias para transformar dados brutos em informação significativa e útil que pode ser utilizada como *insights* e decisões estratégicas, técnicas e operacionais”.

Várias empresas utilizam o *Facebook* como modo de consolidar sua marca e atrair o seu público, a partir da criação de conteúdos relevantes. A maior parte das empresas juniores prezam por utilizar a rede social como ferramenta de prospecção e consolidação da marca, por ser gratuito e trazer resultados positivos. As páginas da rede social contam

¹ <https://nucleocore.org/>

com o *Facebook Insights*, responsável pela visualização de informações sobre os usuários que curtiram a página e o engajamento das postagens realizadas.

As informações disponibilizadas pelo *Facebook Insights* são de extrema importância para a mensuração dos resultados obtidos através de estratégias traçadas pela rede social. A interpretação dessas informações podem ser úteis para as decisões na organização, apesar da plataforma contar com a própria ferramenta de mensuração, a análise e visualização das informações é complexa para pessoas que não possuem conhecimento prévio sobre o assunto, além de ter muitas informações que poderiam ser utilizadas de forma mais eficiente para decisões estratégicas.

O desenvolvimento da plataforma *web* para a análise dos dados obtidos pelas páginas das empresas juniores da Universidade Federal de Ouro Preto, tem como principal objetivo melhorar a visualização e interpretação das informações. Uma vez que, analisar os processos de coleta de informação, criar métricas e indicadores nas mídias sociais e entender os processos para mensuração de resultados são de grande importância para gestão de marca, produtos e relacionamento com seu público (PALOMINO; ANDRADE, 2013).

1.2 Objetivos

O objetivo geral deste trabalho é o desenvolvimento de uma plataforma *web* para monitoramento das páginas do *Facebook* das empresas juniores da Universidade Federal de Ouro Preto. As informações serão extraídas por meio da *Application Programming Interface* (API) da rede social, e serão aplicadas técnicas de mineração de dados, marketing e desenvolvimento *web*. O propósito é melhorar a maneira de visualização das informações obtidas e apoiar a tomada de decisão.

Este trabalho possui aos seguintes objetivos específicos:

1. Modelar e implementar uma aplicação *web* para monitoramento das redes sociais das empresas juniores.
2. Coletar dados por meio da API do *Facebook*.
3. Propor relatórios que auxiliem na análise das *fan pages* de modo descomplicado.

1.3 Metodologia

As etapas para a elaboração deste projeto foram definidas da seguinte maneira:

1. Revisar a literatura: estudar os conceitos e aplicações de técnicas de mineração de dados que auxiliarão no desenvolvimento, compreender a API do *Facebook* para

que o acesso as informações sejam feitas de maneira segura e correta, dinamizar a interpretação dos dados e estudar trabalhos correlatos e identificar pontos de melhorias que a plataforma proposta pode oferecer.

2. Definir um modelo: desenvolver a documentação do software, abrangendo os recursos disponibilizados pela [API](#) do *Facebook*.
3. Estudar representações: buscar formas de visualização de dados que possam auxiliar na exibição de resultados e realizar a prototipação do software, tendo como principal objetivo o desenvolvimento de forma clara.
4. Implementar o software: desenvolvimento do código fonte aplicando boas práticas de programação.
5. Analisar e discutir os resultados obtidos, além de identificar possíveis melhorias e considerações gerais sobre a aplicação.

1.4 Organização do trabalho

O restante do trabalho é organizado, conforme descrito a seguir. No [Capítulo 2](#) será apresentada a revisão bibliográfica, as ferramentas e tecnologias utilizadas e os trabalhos correlatos. O escopo acerca do projeto é apresentado no [Capítulo 3](#). Já no [Capítulo 4](#) será retratado a implementação da plataforma e seus resultados. Por fim, no [Capítulo 5](#) são apresentadas as considerações finais e as propostas para trabalhos futuros.

2 Revisão bibliográfica

Nesse capítulo será apresentada a revisão bibliográfica contendo conceitos de mineração de dados em redes sociais, as ferramentas aplicadas ao longo do desenvolvimento do sistema. Além disso, serão apresentados e discutidos os trabalhos correlatos.

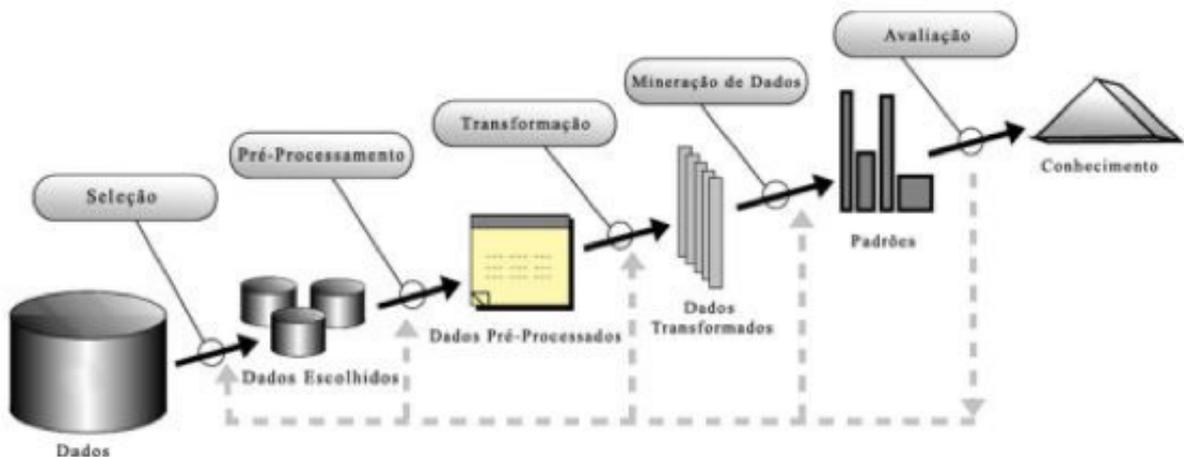
2.1 Mineração de dados em Redes Sociais

Diariamente, uma enorme quantidade de dados é produzida por meio das redes sociais, esse fato advém da popularização das redes sociais. Os usuários por meio de interações compartilham e recebem muita informação, influenciando e sendo influenciados pelos seus amigos (CHA et al., 2010).

A análise e extração de dados obtidos das redes sociais podem ser aplicados em diversas áreas, como ciências comportamentais e *marketing* (WASSERMAN; FAUST et al., 1994). O conhecimento obtido por meio da interpretação dos dados auxilia diretamente na tomada de decisões e na vantagem competitiva para as organizações (FREITAS et al., 2008).

A mineração de dados é uma das etapas do processo denominado *Knowledge Discovery in Databases* (KDD) ou Descoberta de Conhecimento em Banco de Dados. As principais etapas do KDD, segundo Fayyad (1996), podem ser visualizadas na Figura 1:

Figura 1 – Processo de KDD



Fonte: Retirado de Silva (2017)

De acordo com a Fayyad (1996) as etapas podem ser caracterizadas como:

1. **Seleção:** são selecionados os dados significantes a serem utilizados no processo de busca de padrões e geração de conhecimento.
2. **Pré-processamento:** os dados passam por um processo de limpeza. Dados inconsistentes, redundantes e inválidos são descartados.
3. **Transformação:** aplicação de técnicas para para transformação dos dados de forma adequada, para ser trabalhado na mineração de dados.
4. **Mineração de dados:** busca de padrões sobre os dados através da utilização de algoritmos que satisfazem o objetivo proposto, fazendo com que as informações importantes ajudem no processo de tomada de decisão.
5. **Avaliação:** análise de resultados proveniente da mineração de dados e geração de conhecimento resultando em benefícios para a organização.

Este trabalho aplica o método de mineração de dados em uma *fan page* do *Facebook* para analisar a performance e traçar padrões que agreguem nas tomadas de decisão.

2.2 Framework

A utilização de *frameworks* em projetos de desenvolvimento de *software* tem se tornado cada vez maior, devido a necessidade de um projeto de qualidade em pouco tempo. O uso correto de um *framework* reduz consideravelmente o tempo implementação do sistema (VALLE, 2019).

Compreende-se como um *framework* um conjunto de classes ou funções implementadas em uma linguagem de programação específica que visam auxiliar no desenvolvimento de *software* (GABARDO, 2017). A organização do trabalho em um *framework* ocorre a partir da utilização do padrão *Model-View-Controller* (MVC). As requisições do *software* são direcionadas para a camada *Controller*, que por sua vez acessa a camada *Model* responsável pelo processamento das requisições e por fim, os resultados são exibidos na camada *View* (LUCIANO; ALVES, 2017), possibilitando um projeto padronizado e organizado.

Durante o desenvolvimento do *software* foi utilizado o *framework Laravel*¹. O *Laravel* é um *framework open source*, que utiliza da linguagem de programação *PHP: Hypertext Preprocessor* (PHP) na construção de sistemas *web*.

¹ <https://laravel.com/>

2.3 *Laravel Charts*

O *Laravel Charts*² é uma biblioteca disponível para o *Laravel* responsável por gerar gráficos. A biblioteca foi projetada para ser extensível e personalizável, sendo possível utilizar de bibliotecas do *JavaScript*, como o *Chart.js*³ para visualização de dados por meio de recursos gráficos.

O recurso foi utilizado para auxiliar visualização da análise de uma *fan page*. O *Laravel* é responsável pela recuperação de dados da base de dados da plataforma e o *Laravel Charts* garante a visualização dos resultados por meio de gráficos.

2.4 Banco de Dados

O banco de dados é um sistema responsável por armazenar informações e permitir que sejam feitas buscas e atualizações sempre que solicitado (DATE, 2004). Para auxiliar na manutenção de um banco de dados é possível utilizar um sistema de gerenciamento de banco de dados, responsável por gerenciar o acesso, manipular e organizar os dados.

Durante o projeto foi utilizado o *MySQL Workbench*⁴, uma ferramenta visual para a criação do banco de dados. O *MySQL Workbench* é uma ferramenta de simples utilização e intuitiva. Com ela é possível modelar os dados e desenvolver em *Structured Query Language (SQL)*.

2.5 Python

O *Python*⁵ é uma linguagem de programação *open source*, com a sintaxe clara e sucinta tornando o processo de desenvolvimento de *software* mais produtivo e o código-fonte mais legível (BORGES, 2014). No desenvolvimento do projeto foi utilizado o *Python* como linguagem *script*, que permite a automatização de tarefas e conseqüentemente reduzindo o tempo de implementação.

2.6 Graph API

Atualmente o *Facebook* é uma das redes sociais mais utilizadas mundialmente sendo um dos principais veículos de consumo de informações. A análise dos dados obtidos através da plataforma é fundamental para tomar decisões e elaborar estratégias eficazes de *marketing* (ALVES, 2016).

² <https://charts.erik.cat/>

³ <https://www.chartjs.org/>

⁴ <https://www.mysql.com/products/workbench/>

⁵ <https://www.python.org/>

A principal maneira de recuperar e gerenciar dados do *Facebook* é a partir da *Application Programming Interface* (API), denominada Graph API ⁶. A partir dela foi possível recuperar os dados provenientes da *fan page* a ser analisada.

2.7 Trabalhos correlatos

Nesta seção serão apresentados dois sistemas correlatos ao *software* desenvolvido. O critério para a escolha desses trabalhos foi a similaridade entre a aplicação o projeto desenvolvido.

2.7.1 Facebook Insights

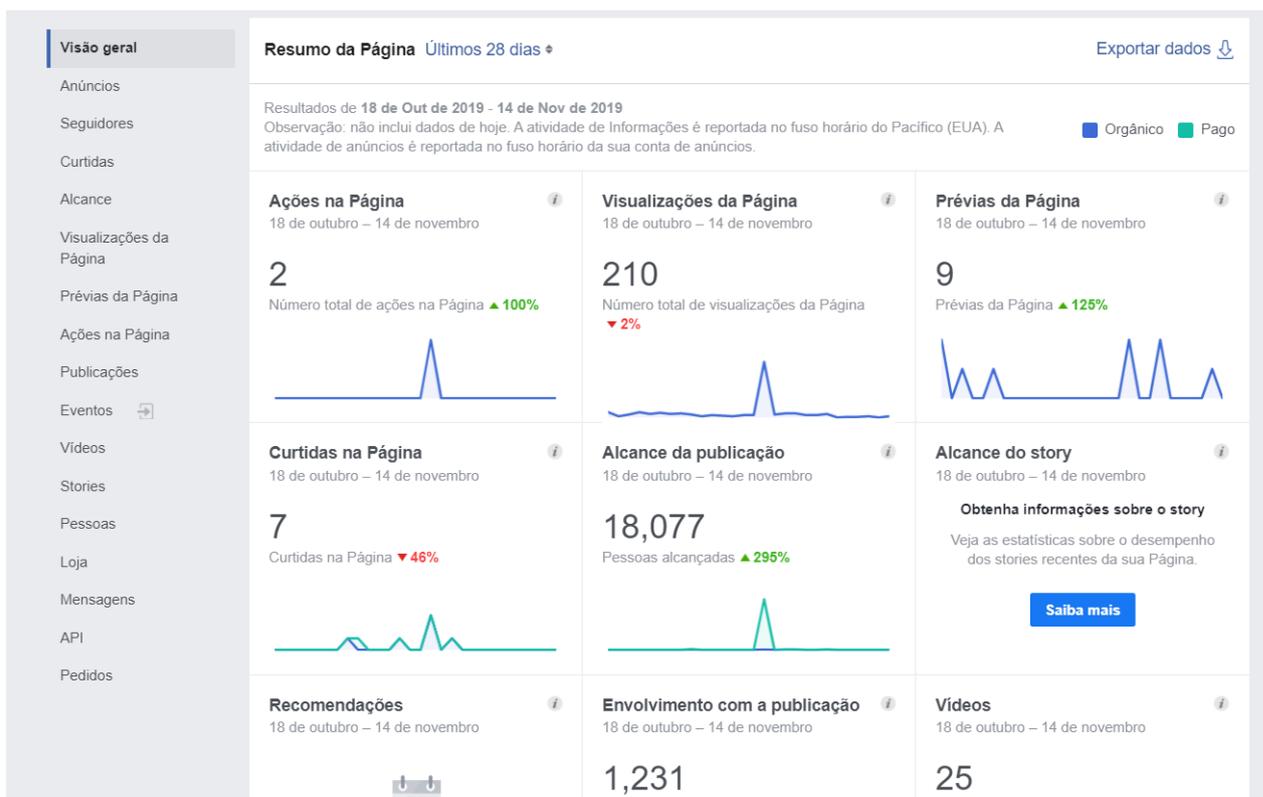
O *Facebook Insights* é a ferramenta nativa e gratuita disponibilizada pelo *Facebook* que permite acompanhar a performance das *fan pages* (AMARAL, 2019). Por meio dela é possível selecionar períodos específicos para análise dos dados: do dia corrente, do dia anterior, de uma semana atrás ou dos últimos 28 dias.

A categorização dos dados é feita a partir das métricas, um sistema de mensuração que tem como objetivo quantificar uma característica, dinâmica ou tendência (FARRIS et al., 2013). É possível visualizar o desempenho de diversas métricas como as curtidas, engajamento e impressões, por exemplo.

A plataforma deixa a desejar na visualização das informações, pois as métricas são informadas sem antes haver um esclarecimento delas e sua visualização não é intuitiva. A Figura 2 ilustra a página inicial da ferramenta, composta por muitas informações sem uma descrição para cada métrica e um detalhamento mais preciso sobre as datas da coleta, podendo ocasionar confusão na interpretação das informações.

⁶ <https://developers.facebook.com/docs/graph-api>

Figura 2 – Facebook Insights - Página inicial



Fonte: Retirado de Facebook (2019a)

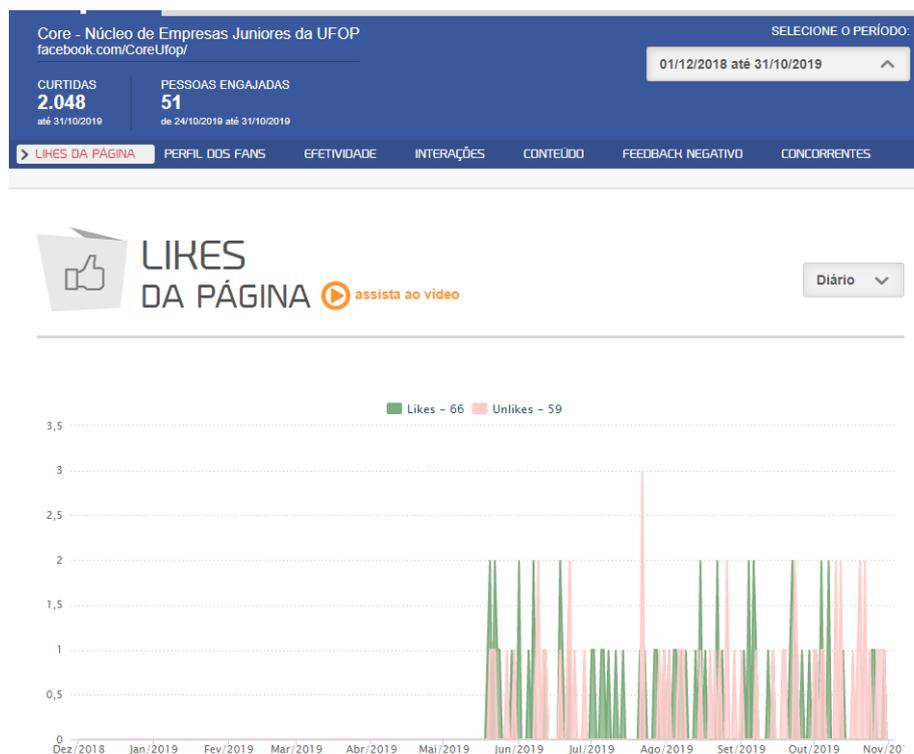
2.7.2 mLabs

O *mLabs*⁷ é um gerenciador de redes sociais, que garante relatórios de monitoramento. Por ser um *software* com assinatura mensal, para avaliar o seu funcionamento, foram realizadas experiências de usuário na versão de testes.

Pôde-se observar que a aplicação garante bons relatórios e auxilia no controle das redes sociais. A visualização das informações é mais evidente e objetiva. Existe uma explicação resumida sobre cada métrica apresentada tornando a interpretação mais fácil.

O sistema apresentado consegue realizar a recuperação dos dados em até seis meses anteriores a data atual. Essa limitação pode prejudicar uma análise mais detalhada da performance da *fan page*. A Figura 3 ilustra a performance da *fan page* em relação as curtidas da página. Observamos que o gráfico utilizado para a visualização pode tornar a interpretação complexa.

⁷ <https://www.mlabs.com.br/>

Figura 3 – *mLabs* - Página inicial

Fonte: Retirado de [mLabs \(2019\)](#)

2.7.3 Comparações com a aplicação proposta

Conforme os trabalhos correlatos apresentados, existem no mercado outros *softwares* responsáveis pelo monitoramento das redes sociais, em especial das *fan pages* das organizações. O *Facebook Insights* possui muitas métricas atreladas a sua plataforma e a visualização das informações é complexa, dificultando o entendimento de pessoas que não possuem um conhecimento prévio sobre o assunto.

A plataforma *mLabs* busca uma interpretação das informações mais intuitiva, por possuir apenas métricas relevantes e uma visualização mais intuitiva focando no entendimento das métricas. Em contrapartida para a utilização do sistema deve ser feito uma assinatura mensal a partir de R\$ 19,90 para ter acesso aos relatórios.

Tendo em vista as características observadas nos trabalhos correlatos citados acima, o objetivo da implementação do *Rocket* foi a utilização da plataforma para membros de empresas juniores, auxiliando na interpretação de dados oriundos de redes sociais. De acordo com [Nascimento e Ferreira \(2005\)](#) a facilidade na visualização dos resultados visa auxiliar no entendimento de um determinado assunto, sem uma visualização intuitiva, exigiria um maior esforço para compreensão.

2.8 Considerações finais

Nesse capítulo foi apresentado a revisão bibliográfica contendo conceitos de mineração de dados e as ferramentas e tecnologias a serem utilizadas. Além disso, sistemas correlatos foram analisados com foco em suas funcionalidades. E por último, foi apresentado a diferença e melhorias propostas pela plataforma *web* a ser desenvolvida. No Capítulo 3 será apresentado o escopo e as funcionalidades do sistema proposto.

3 Modelagem da aplicação

Este capítulo descreve a documentação do projeto. São apresentados a definição do escopo e o problema, as ferramentas e tecnologias utilizadas, a coleta de dados e suas métricas. E por fim, o banco de dados é apresentado.

3.1 Definição do escopo e do problema

Para realizar o desenvolvimento do projeto foi feita a revisão bibliográfica e o escopo, identificando suas necessidades, as ferramentas e tecnologias que foram aplicadas no projeto. Foi realizado um estudo sobre os trabalhos correlatos e a [API](#) disponibilizada pelo *Facebook* para a coleta de dados, afim de obter resultados satisfatórios durante o desenvolvimento da aplicação.

3.2 Estudo de ferramentas e tecnologias

Nessa seção serão apresentadas as ferramentas e tecnologias utilizadas para o desenvolvimento da plataforma *web*. Por ser uma aplicação que requer alimentação contínua de dados por meio de uma [API](#) e precisa estar sempre disponível para o usuário acessar de qualquer lugar com acesso à *internet*, a plataforma *web* atende as necessidades. As outras ferramentas e tecnologias utilizadas no projeto são citadas a seguir:

1. **Plataforma:** *web*;
2. **Servidor:** *Apache 2.4*;
3. **Linguagens:** *PHP 7.1.3*, *HTML 5*, *Python 3.7.4*, *Javascript*, *JavaScript Object Notation (JSON)*, *Cascading Style Sheets (CSS)*;
4. **Framework:** *Laravel 5.7*;
5. **API:** *Graph API*¹;
6. **Servidor do banco de dados:** *MySQL Community Server 8.0.12*;
7. **Template *HTML*:** *Material Dashboard*²
8. **Gerenciador de controle de versão:** *GitHub*³

¹ <https://developers.facebook.com/docs/graph-api/>

² <https://www.creative-tim.com/product/material-dashboard-laravel>

³ <https://github.com/>

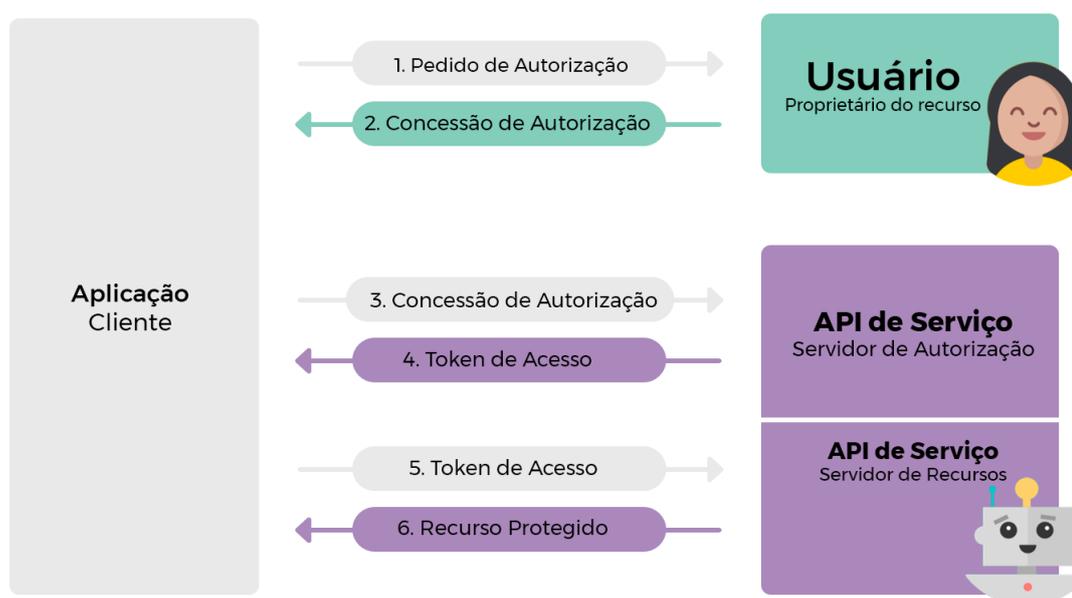
9. Ambiente de desenvolvimento: *Visual Code Studio*⁴

3.3 Coleta de dados no *Facebook*

A aplicação de técnicas de mineração de dados pode fazer com que as organizações podem delinear o comportamento *online* de potenciais clientes, atingindo seu público ideal (HIRATA, 2014). A extração de dados das *fan pages* do *Facebook* é realizada com o auxílio da *Graph API* disponibilizada pelo próprio *Facebook*.

A *Graph API* é uma *API* baseada em uma conexão *Hypertext Transfer Protocol* (*HTTP*), na qual as solicitações são enviadas ao *Uniform Resource Locator* (*URL*) de hospedagem da *API* e, em seguida, retorna os resultados (FACEBOOK, 2019b). Para ter acesso aos dados provenientes da *Graph API*, é necessário registrar uma aplicação a partir do *Facebook for Developers*⁵. O registro fornece uma série de chaves e *tokens* que são utilizados no *script* da aplicação, além da autorização por parte do usuário. A Figura 4 ilustra detalhadamente o fluxo de comunicação realizadas pelo protocolo *OAuth 2.0*.

Figura 4 – Fluxo *OAuth 2.0* para obtenção das chaves e *tokens* de acesso a *Graph API*



Fonte: adaptado de Slack (2019)

De acordo com Slack (2019) as etapas ilustradas na Figura 4 podem ser descritas como:

1. A aplicação envia para o usuário um pedido de autorização, informando quais informações gostaria de utilizar;

⁴ <https://code.visualstudio.com/>

⁵ <https://developers.facebook.com/>

2. Caso o usuário autorize o aplicativo é redirecionado de volta uma chave de acesso à aplicação ;
3. A aplicação então utiliza da chave de acesso para conectar com a [API](#) de serviço;
4. A [API](#) de serviço concede o *token* de acesso de informações para a aplicação;
5. A aplicação então faz a requisição dos recursos almejados para a [API](#) de Serviço;
6. A [API](#) concede a aplicação os dados requisitados.

O retorno advindas das requisições da *Graph API* são trocadas no formato [JSON](#) considerado simples para a leitura e escrita humana e computacional ([SILVA, 2017](#)).

3.3.1 Métricas

As métricas são um sistema de mensuração que quantifica uma tendência, característica ou dinâmica ([FARRIS et al., 2013](#)). Para o desenvolvimento da aplicação que tem como objetivo a mensuração da performance das *fan pages* e suas publicações, foram selecionadas algumas métricas disponibilizadas pela *Graph API*. A [Tabela 1](#) descreve as métricas utilizadas para obtenção de dados da *fan page*:

Tabela 1 – Métricas da *fan page*

Métricas	Descrição
<i>page_fans</i>	O número total de pessoas que curtiram a <i>fan page</i>
<i>page_impressions</i>	O número de vezes que qualquer conteúdo da <i>fan page</i> ou sobre ela apareceu na tela de uma pessoa.
<i>page_impressions_by_age_gender_unique</i>	O número de vezes que qualquer conteúdo da <i>fan page</i> ou sobre ela apareceu na tela de uma pessoa, agrupado por gênero e faixa etária.
<i>page_post_engagements</i>	O número de vezes que as pessoas interagiram de alguma forma com as publicações da <i>fan page</i> por meio de curtidas, compartilhamentos, comentários entre outros.
<i>page_actions_post_reactions_like_total</i>	Total de curtidas nas publicações da <i>fan page</i>
<i>page_actions_post_reactions_love_total</i>	Total de reações “Amei” nas publicações da <i>fan page</i>
<i>page_actions_post_reactions_wow_total</i>	Total de reações “Uau” nas publicações da <i>fan page</i>
<i>page_actions_post_reactions_haha_total</i>	Total de reações “Haha” nas publicações da <i>fan page</i>
<i>page_actions_post_reactions_sorry_total</i>	Total de reações “Triste” nas publicações da <i>fan page</i>
<i>page_actions_post_anger_haha_total</i>	Total de reações “Grr” nas publicações da <i>fan page</i>
<i>page_actions_post_reactions_total</i>	Total de reações nas publicações da <i>fan page</i> categorizadas por tipo

Fonte: elaborado pelo autor

Para obter as métricas das postagens da *fan page* é necessário recuperar as postagens inicialmente. A [Tabela 2](#) apresenta os parâmetros utilizados para a recuperação das postagens:

Tabela 2 – Parâmetros das publicações

Parâmetros	Descrição
<i>id</i>	O identificador único (id) da postagem.
<i>type</i>	O tipo da postagem.
<i>created_time</i>	A hora em que a postagem foi publicada.
<i>full_picture</i>	A imagem contida na postagem.
<i>message</i>	A legenda da postagem

Fonte: elaborado pelo autor

Após recuperar as postagens é possível acompanhar sua performance. A [Tabela 3](#) apresenta as métricas utilizadas para análise das postagens:

Tabela 3 – Métricas das publicações

Métricas	Descrição
<i>post_impressions</i>	O número de vezes que a publicação foi exibida na tela de uma pessoa.
<i>post_impressions_fan</i>	O número de vezes que a publicação foi exibida na tela de uma pessoa que segue a <i>fan page</i> .
<i>post_engaged_fan</i>	O número de pessoas que curtem a <i>fan page</i> e interagiram com a publicação.
<i>post_reactions_like_total</i>	Total de curtidas na publicação.
<i>post_reactions_love_total</i>	Total de reações “Amei” na publicação.
<i>post_reactions_wow_total</i>	Total de reações “Uau” na publicação.
<i>post_reactions_haha_total</i>	Total de reações “Haha” na publicação.
<i>post_reactions_sorry_total</i>	Total de reações “Triste” na publicação.
<i>post_reactions_anger_total</i>	Total de reações “Grr” na publicação.

Fonte: elaborado pelo autor

A [Figura 5](#) ilustra a transição de informações que ocorre na aplicação:

Figura 5 – Transição das informações



Fonte: elaborado pelo autor

As etapas exemplificadas na [Figura 5](#) podem ser descritas como:

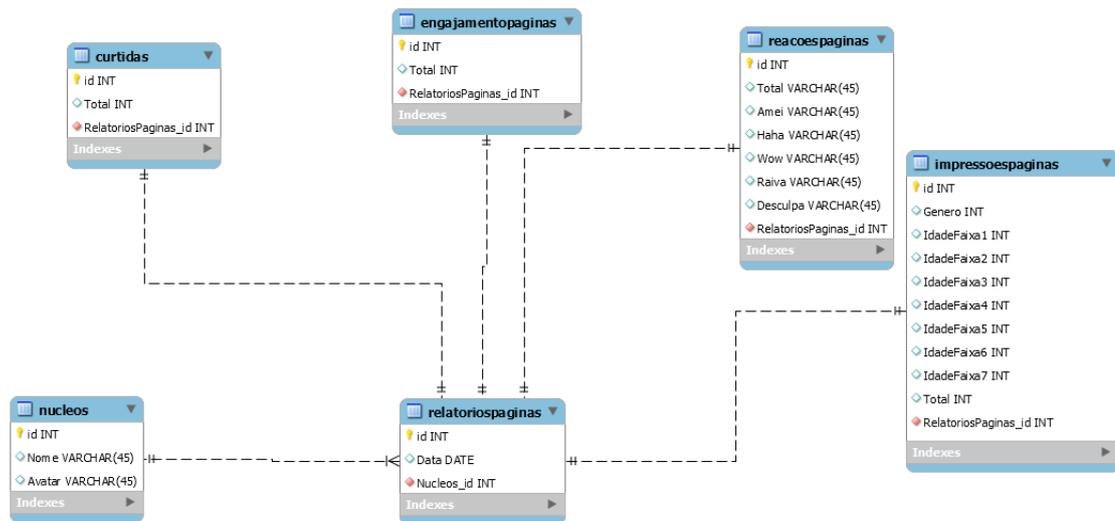
1. A aplicação coleta os dados das métricas definidas na Seção 3.3, por meio de um *script*;
2. As informações recuperadas são armazenados na base de dados da plataforma ;
3. A base de dados é analisada e o resultado é exibido para os usuários na aplicação, por meio de recursos gráficos facilitando a compreensão.

3.4 Banco de dados

Logo após a definição das métricas, foi modelado o banco de dados. O modelo do banco de dados foi definido para armazenar as postagens e as métricas da *fan page* e elaborado para melhorias futuras. Para facilitar a visualização do Diagrama Lógico, o banco de dados separado em dois módulos: relatórios da página e relatórios de postagens.

A Figura 6 mostra a estrutura responsável por armazenar as métricas da *fan page*.

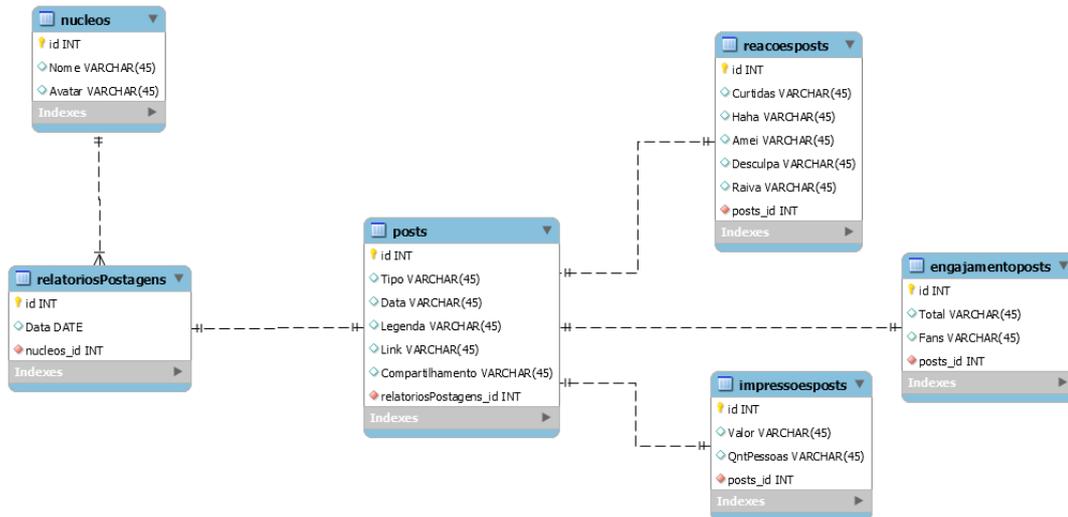
Figura 6 – Diagrama Lógico - Relatórios da *fan page*



Fonte: elaborado pelo autor

A Figura 7 mostra a estrutura responsável por armazenar as postagens e suas métricas.

Figura 7 – Diagrama Lógico - Relatórios de Postagens



Fonte: elaborado pelo autor

A base de dados é responsável por armazenar as métricas definidas na Seção 3.3 extraídas por meio da *Graph API*. O banco de dados completo se encontra no apêndice [Apêndice A](#).

3.5 Considerações finais

Nesse capítulo foi apresentada a parte teórica do desenvolvimento da plataforma por meio do estudo do escopo e as ferramentas e tecnologias apresentadas. No próximo capítulo será abordado a implementação da plataforma e os devidos detalhes.

4 Implementação da aplicação

Neste capítulo serão apresentadas as etapas do processo de desenvolvimento do sistema, bem como o detalhamento de cada passo.

4.1 Desenvolvimento do sistema

O desenvolvimento do sistema consistiu nas seguintes etapas:

1. Criação do banco de dados por meio do *framework Laravel*
2. Desenvolvimento do *script* para coleta dos dados
3. Autenticação dos usuários
4. Página inicial da plataforma
5. Relatório anual da *fan page*
6. Relatório por período da *fan page*
7. Relatório anual das publicações da *fan page*
8. Relatório por período das publicações da *fan page*

Cada uma dessas etapas será descrita a seguir. As considerações sobre cada ação, bem como os passos para execução serão também apresentados.

4.1.1 Banco de dados no *Laravel*

O *framework Laravel* possui um recurso para gerenciamento da estrutura do banco de dados denominado *migration*. Esse recurso permite o controle de versão do banco de dados. Além disso, facilita o envio de modificações realizadas para o banco de dados da aplicação, bem como permite a migração da estrutura para outros SGBDs suportados. Os *migrations* garantem o histórico de alterações realizadas e permitem revertê-las na base de dados.

Após a criação de um *migration*, é necessário executar o comando *php artisan migrate*, para que as modificações sejam enviadas para a base de dados. A [Figura 8](#) ilustra a utilização do *migration* para a criação da tabela "impressões páginas" no banco de dados da aplicação.

Figura 8 – Exemplo de *migration* da tabela impressões páginas

```
public function up()
{
    Schema::create('impressoespaginas', function (Blueprint $table) {
        $table->engine = 'InnoDB';
        $table->increments('id');
        $table->integer('Genero')->nullable();
        $table->integer('IdadeFaixa1')->nullable();
        $table->integer('IdadeFaixa2')->nullable();
        $table->integer('IdadeFaixa3')->nullable();
        $table->integer('IdadeFaixa4')->nullable();
        $table->integer('IdadeFaixa5')->nullable();
        $table->integer('IdadeFaixa6')->nullable();
        $table->integer('IdadeFaixa7')->nullable();
        $table->integer('Total')->nullable();
        $table->integer('RelatoriosPaginas_id')->unsigned();
        $table->index("RelatoriosPaginas_id")->references('id')->on('relatoriosPaginas');
    });
}
```

Fonte: elaborado pelo autor

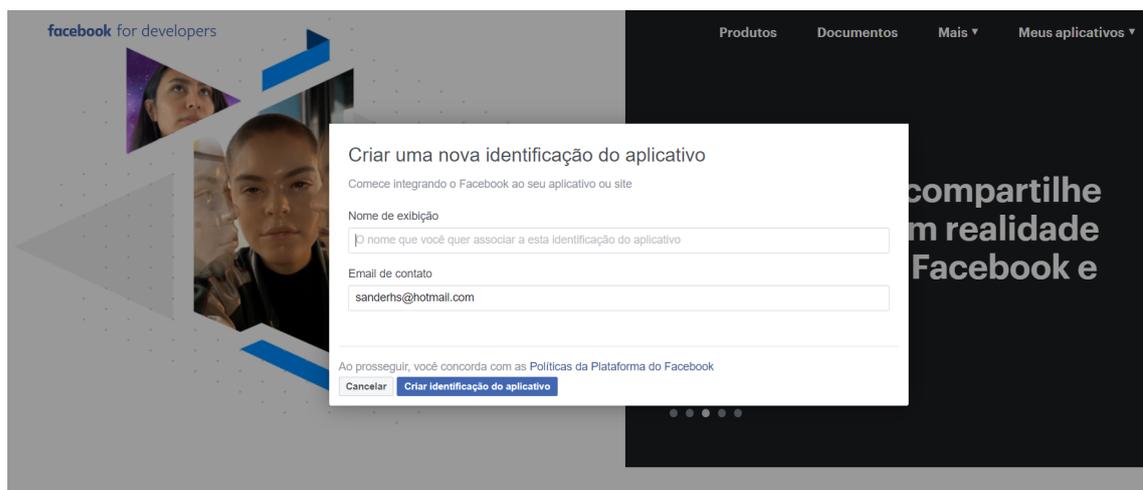
A seguir, o módulo de coleta de dados será apresentado. Ele pode ser utilizado de maneira independente da plataforma do usuário. A coleta é realizada e os dados são armazenados em um banco de dados. Os detalhes da implementação são discutidos adiante.

4.1.2 Coleta dos dados a partir da *Graph API*

Para a coleta de dados foi utilizada a [API](#) chamada *Graph API*, a qual permite a leitura de dados do *Facebook*. A rede social em questão possui uma política de segurança complexa que garante integridade e confiança aos usuários. Consequentemente, alguns passos precisam ser seguidos para que a recuperação de dados seja concluída.

O *Facebook* possui uma plataforma exclusiva para desenvolvedores, o *Facebook Developers*¹. É necessário criar uma aplicação e ser um usuário verificado por meio da plataforma, garantindo assim a permissão para utilizar a [API](#) da rede social. A [Figura 9](#) ilustra a criação de uma aplicação a partir do *Facebook Developers*.

¹ <https://developers.facebook.com/>

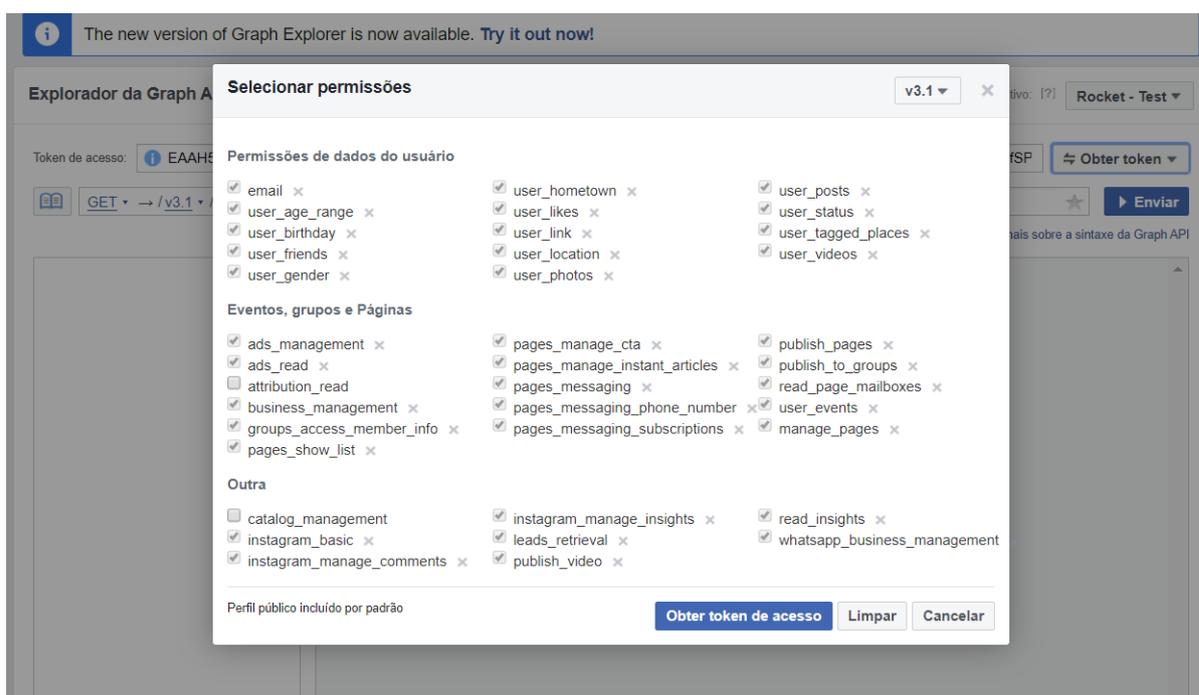
Figura 9 – Criação de uma aplicação a partir do *Facebook Developers*

Fonte: elaborado pelo autor

Como abordado na Seção 3.3, a *Graph API* utiliza da autenticação *OAuth 2.0* e necessita de chaves e *tokens* de acesso do usuário. Após a criação da aplicação, o usuário do *Facebook* precisa conceder acesso a sua conta para a aplicação, e gerar o *token* de acesso para a autenticação. A geração ocorre por meio da ferramenta denominada Explorador da *Graph API*².

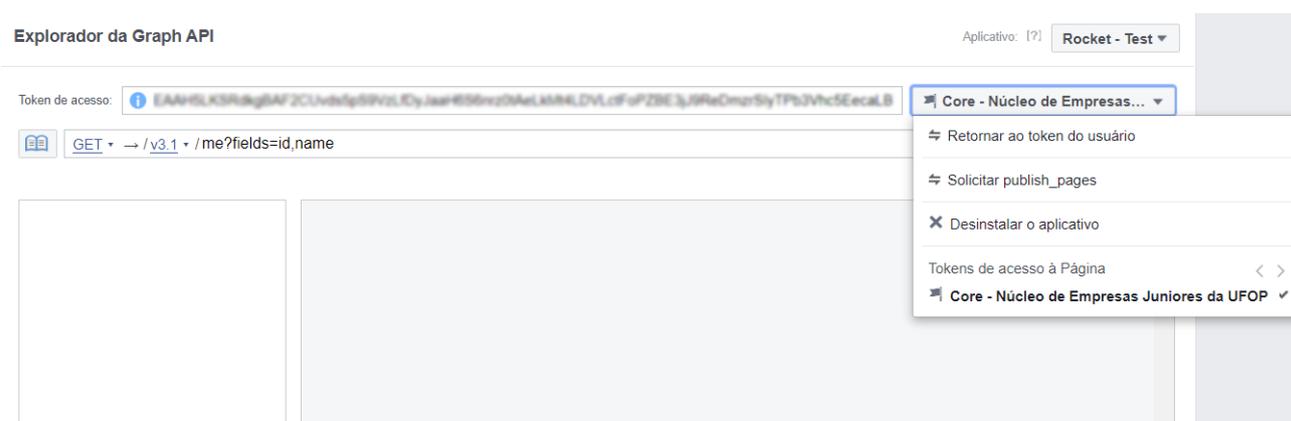
Nessa etapa, é necessário selecionar as permissões que serão concedidas pela *API* por meio do *token* de usuário. A aplicação desenvolvida necessita, em particular, dos dados da *fan page* administrada pelo usuário, por essa razão, dentre as permissões disponibilizadas, a opção de *manage_pages* deve obrigatoriamente ser selecionada. Ela é responsável por recuperar os dados da *fan page* que será utilizada para análise. A Figura 10 ilustra a seleção de permissões para a geração do *token* do usuário.

² <https://developers.facebook.com/tools/explorer/>

Figura 10 – Seleção de permissões para geração do *token* do usuário

Fonte: elaborado pelo autor

Para a implementação do coletor de dados será utilizado a *fan page* do Núcleo de Empresas Juniores da UFOP - Core. É necessário o *token* de permissão por meio da plataforma de desenvolvedor do *Facebook*. O usuário cadastrado na plataforma possui a *fan page* relacionada com sua conta, após selecionar a *fan page* na ferramenta explorador da *Graph API* é possível obter o *token*, como ilustra a [Figura 11](#).

Figura 11 – Obtenção do *token* da *fan page*

Fonte: elaborado pelo autor

A última informação necessária para a iniciar coleta de dados é a chave de segurança e o ID do aplicativo desenvolvido. Ela permite que a *API* faça a conexão entre a aplicação e

a *fan page* utilizada. Por meio do painel de configurações da plataforma *Facebook Developers* é possível recuperá-las, como ilustra a [Figura 12](#).

Figura 12 – Obtenção da chave secreta e ID do aplicativo

The image shows the Facebook Developers application configuration interface. It is a two-column form. The left column includes: 'ID do Aplicativo' (555445111584328), 'Nome de exibição' (Rocket - Test), 'Domínios do aplicativo', 'URL dos Termos de Serviço', and 'Categoria' (Negócio e Páginas). The right column includes: 'Chave Secreta do Aplicativo' (c4c7f1416d796d766d794ca1154b3be1e4) with a 'Redefinir' button, 'Namespace', 'URL da Política de Privacidade', 'Ícone do aplicativo (1024 x 1024)', and 'Uso comercial' (Apoiar minha própria empresa selected).

Fonte: Elaborado pelo autor

Após a obtenção dos *tokens* e chaves de acesso, foi desenvolvido um *script* em *Python* para auxiliar na automatização da coleta de dados. A *API* utilizada é baseada em uma conexão *HTTP*, a qual permite realizar requisições para acesso a dados. Os *tokens* e chaves de acesso obtidos anteriormente foram armazenados no formato *JSON* pela praticidade de manipulação, como ilustra a [Figura 13](#).

Figura 13 – *Tokens* e chave secreta armazenadas no formato *JSON*

```
{
  "usuario": {
    "token": "EAAYLKSADg9dAKL796c7ygg54CZFR6uZCv796d6u4w1T8M9ZCvLX99W6c5Z96P0C7yqZbu38d9p5DZJ4Lw09WuKZAF8geTS"
  },
  "app": {
    "id": "555445111584328",
    "chavesecreta": "c4c7f1416d796d766d794ca1154b3be1e4"
  },
  "page": {
    "token": "EAAYLKSADg9dAKL796c7ygg54CZFR6uZCv796d6u4w1T8M9ZCvLX99W6c5Z96P0C7yqZbu38d9p5DZJ4Lw09WuKZAF8geTS"
    "id": "128319639618128"
  }
}
```

Fonte: elaborado pelo autor

A *Graph API* é instanciada utilizando as informações contidas no arquivo *JSON*. Após a requisição, é retornado um objeto com as permissões anteriormente definidas para a coleta de dados. A [Figura 14](#) exemplifica o código utilizado para realizar a autenticação.

Figura 14 – Código de autenticação da *Graph API*

```
class PaginaApp(object):
    def __init__(self, json_file_name, version):
        super().__init__()

        with open(json_file_name, 'facebook') as f:
            data = json.load(f)
            page_token = data['page']['token']
            self.page_id = data['page']['id']

            self.graph = facebook.GraphAPI(access_token=page_token, version='3.1')
```

Fonte: Elaborado pelo autor

Com a *Graph API* instanciada, é necessário a definição de funções contendo os argumentos para a obtenção dos dados. As funções de recuperação de dados da *fan page* e suas publicações utiliza o objeto da *Graph API*, e têm como parâmetros a identificação da página ou publicação, a métrica a ser analisada, o ano, o mês e o dia. A [Figura 15](#) ilustra a função de relatório de página implementada.

Figura 15 – Função de relatório de página

```
def relatoriopagina(self, metrica, ano, mes, dia):
    return self.graph.get_connections(id=self.page_id,
                                     connection_name='insights',
                                     metric=metrica,
                                     period='day',
                                     since=datetime(ano, mes, dia, 0, 0, 0),
                                     until=datetime(ano, mes, dia, 23, 59, 0))
```

Fonte: elaborado pelo autor

As funções apresentadas anteriormente têm como finalidade a recuperação dos dados fornecidos pelas métricas e o armazenamento no banco de dados. Posteriormente, os dados recuperados foram analisados. Foi desenvolvido no *script* funções para recuperação das métrica definidas na Seção 3.3. Os dados são recuperados em formato **JSON**, tratados e em seguida armazenados na base de dados. Esse processo é representado pela [Figura 16](#).

Figura 16 – Recuperação de curtidas da *fan page*

```
def curtidaspagina(self, ano, mes, dia):  
  
    id= self.relatorio_id  
  
    fans = self.relatoriopagina('page_fans', ano, mes,dia)  
  
    items = fans['data'][0]['values']  
  
    for item in items:  
        curtaida= item['value']  
  
    sql = "INSERT INTO curtidas (total, RelatoriosPaginas_id) VALUES (%s, %s)"  
    val = (curtida,id)  
    cursor.execute(sql, val)  
    conexao.commit()
```

Fonte: elaborado pelo autor

Para recuperar as métricas das publicações da *fan page* é necessário obter as publicações e armazená-las. O identificador único (*id*) da publicação será utilizado como parâmetro para retornar os dados. A [Figura 17](#) ilustra a obtenção das publicações da página e sua adição na base de dados.

Figura 17 – Recuperação de publicações

```
def publicacoes(self, ano, mes,dia):  
    id= self.relatoriopostagens_id  
  
    posts = self.postagens(ano, mes,dia)  
  
    for ind,post in enumerate(posts):  
  
        try:  
            post_id = (post['object_id'])  
        except KeyError:  
            return 0  
        try:  
            data = (post['created_time'][:16])  
        except KeyError:  
            return 0  
        try:  
            tipo = post['type']  
        except KeyError:  
            return 0  
        try:  
            legenda = post['message']  
        except KeyError:  
            return 0  
        try:  
            link = post['full_picture']  
        except KeyError:  
            return 0  
  
    sql = "INSERT INTO posts (Tipo, Data, Legenda, Objeto_id, Link, RelatoriosPostagens_id) VALUES (%s, %s, %s, %s, %s, %s)"  
    val = (tipo,data,legenda,post_id,link,id)  
    cursor.execute(sql, val)  
    conexao.commit()
```

Fonte: Elaborado pelo autor

Logo após definir as funções necessárias para a recuperação dos dados foi implementada a função principal *main*. Ela contém um dicionário com os meses e a quantidade de dias, o ano e a chamada para a execução do processo completo. Isso é representado pela [Figura 18](#).

Figura 18 – Execução do *script*

```
def executar(self, ano, mes, dia):
    self.createrelatoriopagina("{}-{}-{}".format(ano,mes,dia))
    self.createrelatoriopost("{}-{}-{}".format(ano,mes,dia))
    self.posts(ano,mes,dia)
    self.curtidas(ano, mes,dia)
    self.impressoes(ano, mes, dia)
    self.engajamento(ano, mes,dia)
    self.reacoes(ano,mes,dia)

def main():
    app = PaginaApp('facebook.json', version="3.1")
    ano= 2019
    mes=[{"m": 1, "d": 31}, {"m":2, "d":28}, {"m":3, "d":31}, {"m":4, "d":30}, {"m":5, "d":31}, {"m":6, "d":30},
    {"m":7, "d":31}, {"m":8, "d":31}, {"m":9, "d":30}, {"m":10, "d":31}, {"m":11, "d":30}, {"m":12, "d":31}]

    for m in mes:
        for d in range(m["d"]):
            app.executar(ano,m["m"],d+1)

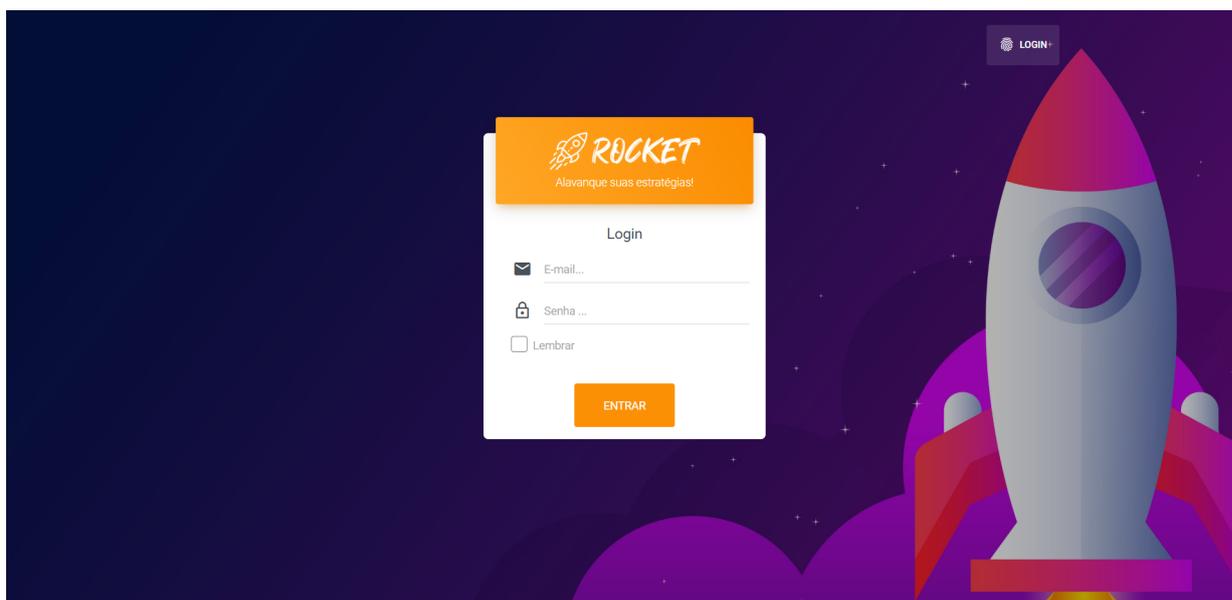
if __name__ == '__main__':
    main()
```

Fonte: elaborado pelo autor

Após a execução da coleta de dados e armazenamento no banco de dados, as informações podem ser apresentadas para o usuário. Os detalhes da plataforma são apresentados a seguir.

4.1.3 Autenticação dos usuários na plataforma

A autenticação dos usuários na plataforma garante a confidencialidade das informações. Para acessar o sistema, o usuário deve informar o seu *e-mail* e a sua senha, como apresentado na [Figura 19](#). Logo após isso, o usuário é redirecionado para a página inicial da plataforma.

Figura 19 – Tela de *login*

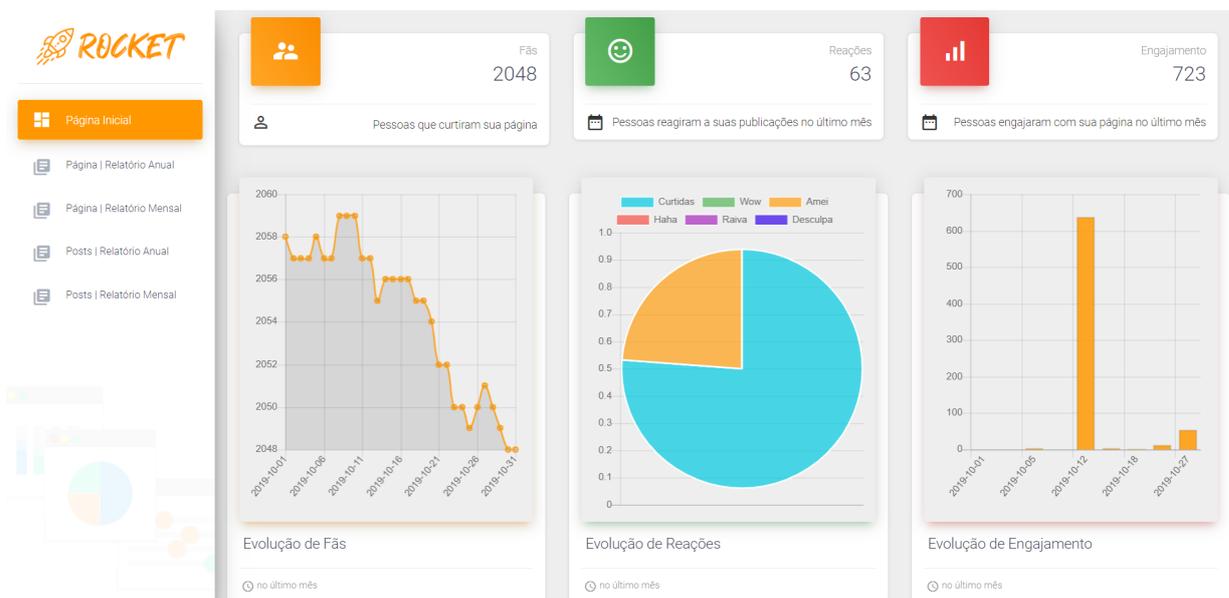
Fonte: elaborado pelo autor

A implementação da autenticação foi realizada por meio do *framework Laravel* e do *template HTML*. Eles possuem recursos para tornarem o processo mais objetivo e mais seguro.

4.1.4 Página inicial da plataforma

A página inicial da plataforma tem como finalidade apresentar para o usuário o monitoramento da *fan page*, contendo o número de fãs, as reações da página, engajamento e a evolução ao decorrer do último mês adicionado na base de dados. A [Figura 20](#) ilustra a página inicial da plataforma.

Figura 20 – Página inicial da plataforma



Fonte: elaborado pelo autor

A recuperação dos dados é feita seguindo o modelo **MVC**, no qual as requisições do *software* são direcionadas para a camada *Controller*. A exibição dos gráficos é realizada com o auxílio da biblioteca *Laravel Charts*, descrita na Seção 2.3. A Figura 21 exemplifica o código utilizado para a seleção dos dados e construção dos gráficos.

Figura 21 – Tratamento dos dados e construção dos gráficos no *Controller*

```

$Ultimomes=DB::table('RelatoriosPaginas')->pluck('data')->last();
$mes= substr($Ultimomes,5,-3);

$curtidas = DB::table('curtidas')->join('RelatoriosPaginas', 'curtidas.RelatoriosPaginas_id', '=', 'RelatoriosPaginas.id')
->whereMonth('data', $mes)->pluck('Total');

$data = DB::table('RelatoriosPaginas')
->whereMonth('data', $mes)
->pluck('Data');

$chart = new CurtidasChart;
$chart->labels($data);
$chart->dataset('', 'line', $curtidas)->color('rgba(255,158,15,0.9)');
$chart->options([
    'legend' =>[
        'display'=> false
    ],
    'scales'=> [
        'yAxes' => [
            'ticks' =>[
                'min'=>2000
            ]
        ]
    ]
])
    
```

Fonte: Elaborado pelo autor

4.1.5 Relatório anual da *fan page*

O relatório anual da *fan page* tem como finalidade obter a evolução das métricas definidas na Tabela 1 da Seção 3.3 durante o último ano da página. É possível visualizar todas as métricas por meio dos gráficos, os quais contêm uma descrição sucinta do conteúdo, a data e o valor recuperado. A Figura 22 ilustra a evolução da métrica *page_fans*, a qual recupera o número total de pessoas que curtiram a *fan page*.

Figura 22 – Evolução anual da métrica *page_fans*



Fonte: elaborado pelo autor

As métricas responsáveis por recuperar as impressões da *fan page* por faixa etária e gênero, e reações da *fan page* por tipo, foram exibidas de maneira diferente das demais para tornar a visualização mais intuitiva e dinâmica. Foi utilizado o gráfico de *donuts* que possibilita visualizar comparações de maneira dinâmica, como ilustra a Figura 23.

Figura 23 – Categorização das impressões por faixa etária e gênero

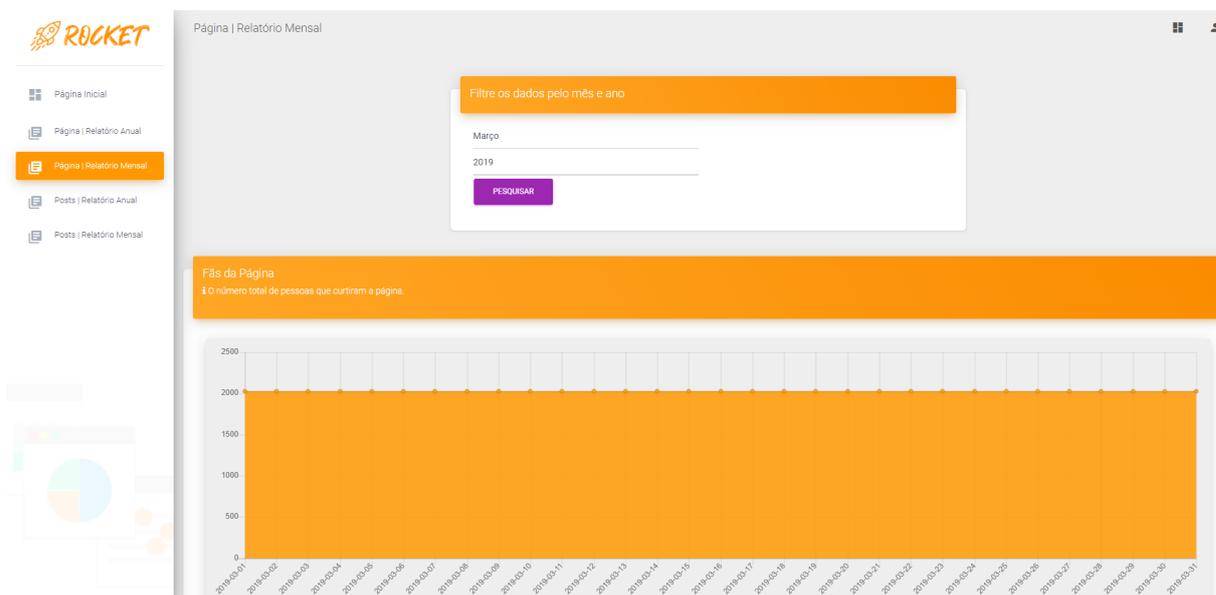


Fonte: Elaborado pelo autor

4.1.6 Relatório por período da *fan page*

O relatório por período da *fan page* tem como finalidade obter a evolução das métricas utilizadas da *fan page* em um intervalo de tempo. Ao acessar a página do relatório, inicialmente o usuário visualiza a evolução das métricas do último mês adicionado na base de dados.

Por meio de um formulário [HTML](#), o usuário tem a opção de selecionar o mês e o ano desejados para análise. A requisição é enviada para ao *Controller* da aplicação responsável por filtrar os dados e retornar as informações solicitadas. A [Figura 24](#) ilustra o relatório por período da *fan page* com o filtro do mês e do ano.

Figura 24 – Relatório por período da *fan page*

Fonte: elaborado pelo autor

4.1.7 Relatório das publicações da *fan page*

O relatório anual das publicações da *fan page* tem como finalidade obter a evolução das métricas definidas na Tabela 2 da Seção 3.3, referente às publicações durante o último ano da *fan page* adicionado na base de dados. É possível visualizar a evolução das métricas por meio dos gráficos.

A Figura 25 ilustra a evolução da métrica *post_engaged_fan*, a qual recupera número de pessoas que curtiram a *fan page* e interagiram com a publicação. O gráfico possui as datas das postagens realizadas e o valor do engajamento em cada delas.

Figura 25 – Evolução anual da métrica *post_engaged_fan*

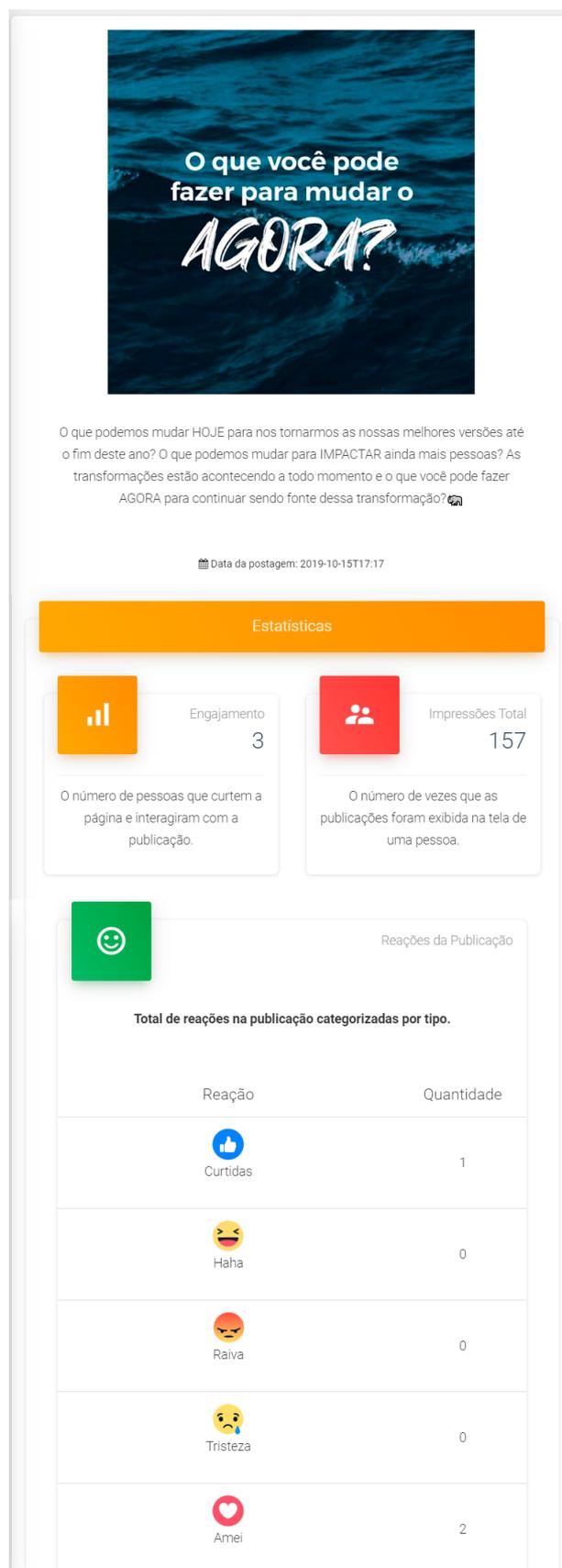


Fonte: elaborado pelo autor

4.1.8 Relatório por período das publicações da *fan page*

O relatório por período das publicações da *fan page* tem como objetivo detalhar as postagens e suas métricas em um intervalo de tempo. Ao acessar a página de relatório mensal das publicações, o usuário consegue visualizar o conteúdo das postagens, o engajamento, a impressão e as reações definidas na Tabela 2 da Seção 3.3. A Figura 26 exemplifica a exibição de cada postagem e as suas respectivas métricas.

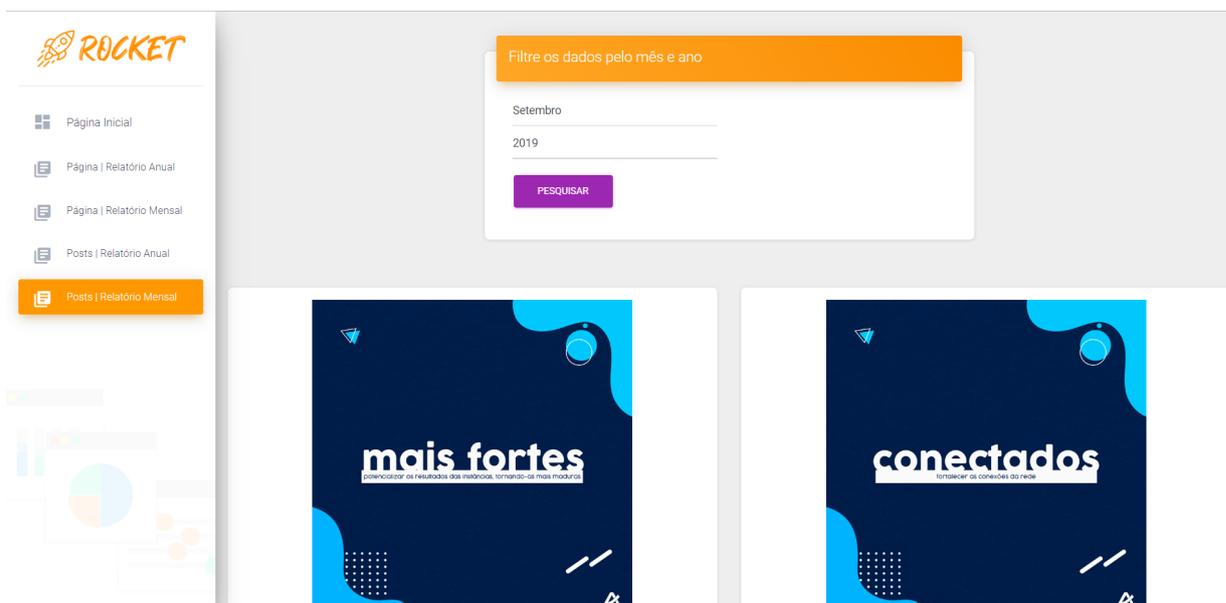
Figura 26 – Exibição da postagem e suas métricas



Fonte: elaborado pelo autor

No primeiro momento são recuperadas as postagens do último mês adicionado na base de dados. A página possui um formulário desenvolvido com [HTML](#) que permite selecionar o mês e o ano disponíveis para recuperação de dados. A [Figura 27](#) ilustra a recuperação de dados a partir do formulário.

Figura 27 – Relatório por período das publicações da *fan page*



Fonte: Elaborado pelo autor

4.2 Avaliações da plataforma

A plataforma desenvolvida tem utilização prática e funcional. Isso se justifica pelo fato de que a plataforma possui funcionalidades que a distingue dos trabalhos correlatos apresentados na [Seção 2.7](#). A plataforma extrai os dados por meio da *Graph API* e armazena na base de dados, permitindo a análise e comparação de dados de diversos períodos desde a sua inicialização. Em contrapartida, os ambientes de análise das *fan pages* do *Facebook* estudados possibilitam a avaliação em um período máximo de três meses anterior a data de consulta.

Por possuir relatórios divididos por evolução anual e por período, contendo métricas da *fan page* e das publicações, o usuário pode ter maior facilidade em encontrar o que necessita. Além disso, as métricas foram descritas na própria plataforma, e a utilização de componentes gráficos para exibir os resultados tornou a plataforma intuitiva e interativa. Ademais, a plataforma é totalmente gratuita, sem necessidade de assinatura mensal para a análise da *fan page*.

O projeto está disponível no GitHub a partir do link: <https://github.com/sanderhs/TCC2>. A aplicação é disponibilizada sob licença *Massachusetts Institute of*

Technology (MIT) e desenvolvida por meio de um *template* livre disponível no site *Creative Tim*³

A plataforma possui algumas limitações, visto que a extração dos dados da *fan page* necessitou de etapas manuais para a definição das chaves e *tokens* de acesso do usuário e da aplicação para a utilização da *Graphi API*. E, por fim, deve-se observar que para o funcionamento correto da plataforma os servidores do *Facebook* devem estar disponíveis.

4.3 Considerações finais

Neste capítulo foram descritos os passos para o desenvolvimento do sistema, os resultados obtidos por meio dos relatórios gerados e a avaliação da plataforma. É importante observar que a maneira de visualização e os filtros definidos tornaram a compreensão das informações intuitivas e detalhadas. No próximo capítulo, as conclusões serão discutidas e as propostas de continuidade do projeto serão apresentadas.

³ <https://www.creative-tim.com/>

5 Conclusão

Este trabalho apresentou o desenvolvimento de uma plataforma para monitoramento das redes sociais das empresas juniores da Universidade Federal de Ouro Preto. Inicialmente, foi apresentada a fundamentação teórica, as tecnologias utilizadas e os trabalhos correlatos. Foi descrita a modelagem da aplicação e o processo de implementação. E, por fim, foram apresentados os resultados.

Durante o processo houve dificuldade na recuperação dos dados a partir da *Graph API*, uma vez que o *Facebook* possui um processo rigoroso para permitir o acesso aos dados. Dessa maneira, a modelagem do problema e implementação se tornaram complexas.

O plataforma tem como propósito gerar relatórios práticos e intuitivos das *fan pages* e suas publicações. Assim sendo, foi fundamental o estudo de representações e a utilização da biblioteca *Laravel Charts* para a implementação dos relatórios. A aplicação do *framework Laravel* no sistema garantiu o tratamento dos dados de maneira menos complexa e tornou o desenvolvimento mais ágil.

5.1 Trabalhos futuros

A versão atual da plataforma necessita que os *tokens* e as chaves de segurança do aplicativo, do usuário e da *fan page* sejam pré-definidos anteriormente à coleta de dados. Para que o usuário consiga conectar sua conta com a aplicação de maneira automática, é necessário uma inspeção criteriosa do *Facebook* para habilitar esta função. Por ser um projeto complexo devido a recuperação de dados, não foi possível tornar automática processo de coleta dos dados. Espera-se que as limitações encontradas sejam desenvolvidas nos trabalhos futuros.

Como propostas de continuidade do projeto foram identificados as seguintes ações principais:

1. Incluir a análise de outras métricas disponíveis na *Graph API*.
2. Incluir outras redes sociais na plataforma para análise além do *Facebook*.
3. Criar funcionalidades para o usuário definir metas em sua *fan page* de acordo com as métricas utilizadas.
4. Criar funcionalidades para o usuário realizar o *download* dos relatórios no formato *Portable Document Format* ([PDF](#)).
5. Implementar testes unitários para validação da aplicação.

Referências

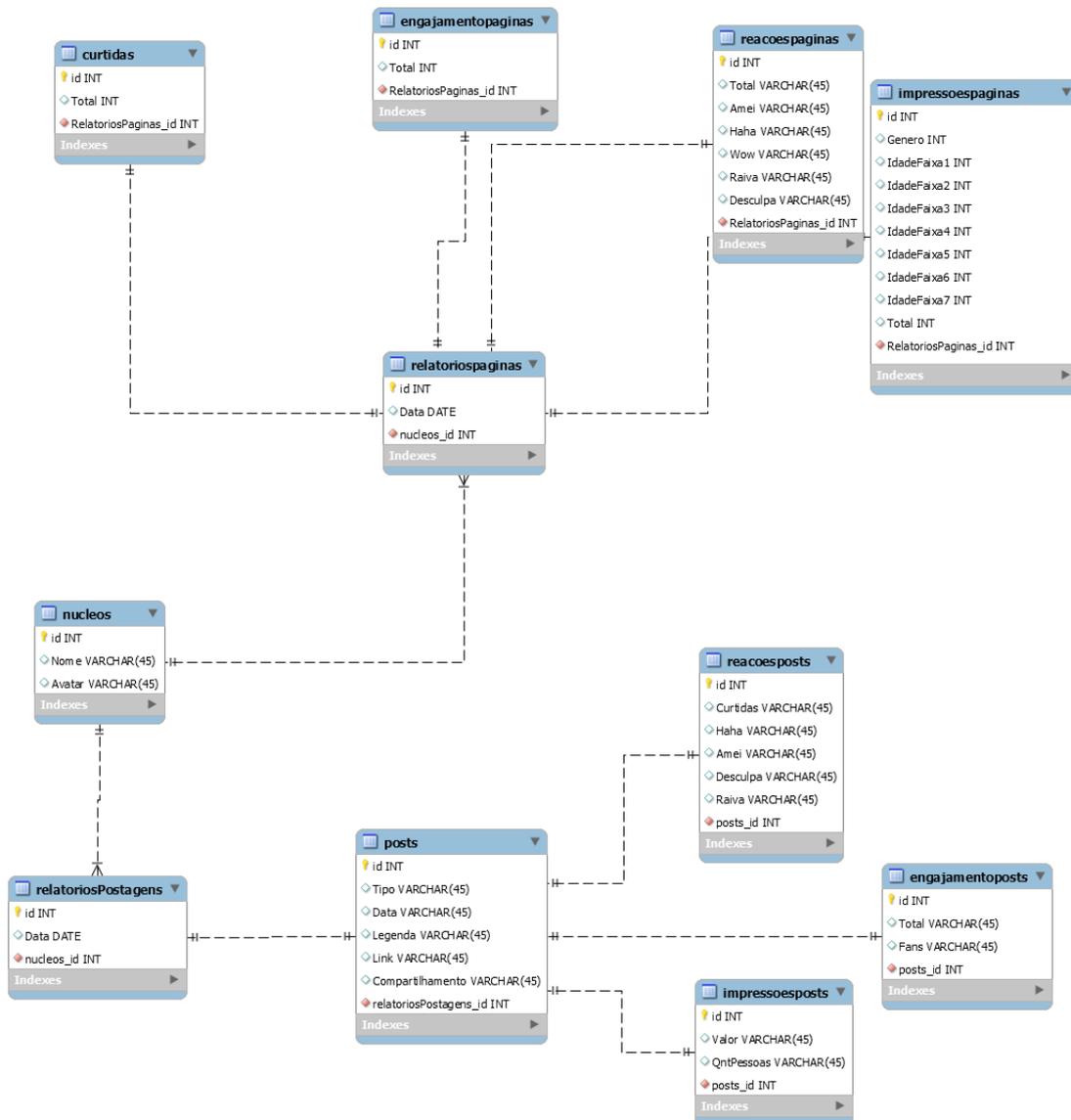
- ALVES, M. Abordagens da coleta de dados nas mídias sociais. *SILVA, T. STABILE, M. Monitoramento e pesquisa em mídias sociais: metodologias, aplicações e inovações. São Paulo: Uva Limão, 2016.* Citado na página 20.
- AMARAL, L. *Facebook Insights: aprenda tudo sobre esse recurso!* 2019. Disponível em: <<https://rockcontent.com/blog/facebook-insights/>>. Citado na página 21.
- AMARAL, P. P. et al. Relações públicas e o gerenciamento nas redes sociais: uma análise das postagens realizadas por empresas de são borja no facebook. Universidade Federal do Pampa, 2015. Citado na página 14.
- BORGES, L. E. *Python para desenvolvedores: aborda Python 3.3.* [S.l.]: Novatec Editora, 2014. Citado na página 20.
- BRASIL. *LEI Nº 13.267, DE 6 DE ABRIL DE 2016 - Disciplina a criação e a organização das associações denominadas empresas juniores, com funcionamento perante instituições de ensino superior.* 2016. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2016/lei/L13267.htm>. Citado na página 14.
- CHA, M. et al. Measuring user influence in twitter: The million follower fallacy. In: *fourth international AAAI conference on weblogs and social media.* [S.l.: s.n.], 2010. Citado na página 18.
- DATE, C. J. *Introdução a sistemas de bancos de dados.* [S.l.]: Elsevier Brasil, 2004. Citado na página 20.
- EVELSON, B. *Topic Overview: Business Intelligence - An Information Workplace Report.* 2008. Disponível em: <<http://www.forrester.com/Topic+Overview+Business+Intelligence/-/E-RES39218?objectid=RES39218>>. Citado na página 15.
- FACEBOOK. 2019. Disponível em: <<https://www.facebook.com/>>. Citado na página 22.
- FACEBOOK. *Graph API Overview.* 2019. Disponível em: <<https://developers.facebook.com/docs/graph-api/overview/>>. Citado na página 26.
- FACEBOOK. *Informações da empresa.* 2019. Disponível em: <<https://br.newsroom.fb.com/company-info/>>. Citado na página 14.
- FARRIS, P. W. et al. *Métricas de Marketing-: O Guia Definitivo de Avaliação do Desempenho do Marketing.* [S.l.]: Bookman, 2013. Citado 2 vezes nas páginas 21 e 27.
- FAYYAD. The kdd process for extracting useful knowledge from volumes of data. *Communications of the ACM*, ACM, v. 39, n. 11, p. 27–34, 1996. Citado na página 18.
- FERREIRA, G. C. Information social networks: one history and one case study. *Perspectivas em Ciência da Informação*, SciELO Brasil, v. 16, n. 3, p. 208–231, 2011. Citado na página 14.

- FREITAS, C. et al. Extração de conhecimento e análise visual de redes sociais. *SEMISH-Seminário Integrado de Software e Hardware, Belém do Pará, Brasil, SBC*, p. 106–120, 2008. Citado na página 18.
- GABARDO, A. C. *Laravel para ninjas*. [S.l.]: Novatec Editora, 2017. Citado na página 19.
- HIRATA, A. O facebook e o direito à privacidade. *id/502970*, 2014. Citado na página 26.
- IBGE. *Pesquisa Nacional por Amostra de Domicílios Contínua - Trimestral*. 2018. Disponível em: <https://ww2.ibge.gov.br/home/estatistica/indicadores/trabalhoerendimento/pnad_continua/default.shtm>. Citado na página 14.
- JUNIORES, C. B. de E. 2019. Disponível em: <<https://brasiljunior.org.br/conheca-o-mej>>. Citado 2 vezes nas páginas 14 e 15.
- LUCIANO, J.; ALVES, W. J. B. *PADRÃO DE ARQUITETURA MVC: MODEL-VIEWCONTROLLER*. 2017. Citado na página 19.
- MLABS. *Plataforma de Gestão de Redes Sociais e Agendamento de Posts*. 2019. Disponível em: <<https://www.mlabs.com.br/>>. Citado na página 23.
- NASCIMENTO, H. A. D.; FERREIRA, C. B. Visualização de informações—uma abordagem prática. In: *XXV Congresso da Sociedade Brasileira de Computação, XXIV JAI. UNISINOS, S. Leopoldo-RS*. [S.l.: s.n.], 2005. Citado na página 23.
- PALOMINO, P. T.; ANDRADE, L. A. de. Perspectivas em redes sociais: a inteligência coletiva como ferramenta de análise de métricas e indicadores de desempenho. *Revista GEMInIS*, v. 4, n. 2, p. 164–175, 2013. Citado na página 16.
- REZENDE, S. O. et al. Mineração de dados. *Sistemas inteligentes: fundamentos e aplicações*, v. 1, p. 307–335, 2003. Citado na página 15.
- SANTOS, A. D. P. d. Descobrimos eventos locais utilizando análise de séries temporais nos dados do twitter. 2013. Citado na página 15.
- SILVA, M. M. Mineração de dados no twitter: uma ferramenta prática para extração e análise dos resultados. 2017. Citado 2 vezes nas páginas 18 e 27.
- SLACK. *Using OAuth 2.0*. 2019. Disponível em: <<https://api.slack.com/docs/oauth>>. Citado na página 26.
- STATISTA. *Global social media ranking 2019*. 2019. Disponível em: <<https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/>>. Citado na página 14.
- VALLE, T. S. Sistema web para gerenciamento de prontuário do paciente. 2019. Citado na página 19.
- WASSERMAN, S.; FAUST, K. et al. *Social network analysis: Methods and applications*. [S.l.]: Cambridge university press, 1994. v. 8. Citado na página 18.

Apêndices

APÊNDICE A – Diagrama do Banco de Dados

Figura 28 – Diagrama completo do Banco de Dados



Fonte: Elaborado pelo autor