

PAINEL DE DADOS FEBRACE: **IDENTIFICAÇÃO GEOGRÁFICA DE ESCOLAS PREMIADAS(Backend)**

Richard Matheus Bezerra Ataliba
Josenalde Barbosa de Oliveira

RESUMO

O projeto consiste em organizar as listas de premiados da Feira Brasileira de Ciências e Engenharia (FEBRACE). A feira todos os anos desde 2003 disponibiliza dois PDFS, uma lista de projetos finalistas e outra de projetos premiados(FEBRACE,2023).

Em ambos os pdfs há dados relacionais, como o nome de projetos premiados, que também pode ser acessado no PDF de finalistas. Porém o colégio das equipes premiadas não pode ser acessado diretamente pelo PDF de premiados, já que esse dado só existe no PDF de finalistas, para isso seria necessário pesquisar os campos relacionais (Nome do projeto ou Integrantes) no PDF de finalistas, e então acessar qual a escola daquele determinado projeto. Estatísticas de por exemplo, escola com maior taxa de premiados, ou estados com maiores premiados, ou várias outras informações que poderão ser acessadas no decorrer do projeto, pode mostrar a qualidade de ensino daquele colégio ou região , e como determinados projetos e métodos de ensino podem influenciar e contribuir para as competências de alunos nas áreas das ciências.

Palavras chave: Ciência de dados; Mineração de textos; MongoDB; Python

1 INTRODUÇÃO

O conteúdo disponibilizado pela FEBRACE(feira brasileira de ciências e engenharias) pode ser acessado por qualquer usuário comum através do site oficial do órgão (FEBRACE,2023). Esse conteúdo se baseia em dois principais arquivos no formato .PDF para cada ano, um arquivo com a lista de projetos premiados e outro com a lista de projetos finalistas no respectivo ano.

Entretanto, a plataforma se prende apenas a esse único formato no que se diz respeito a visualização de seus dados; divisões por região, categorias, ou quantidade de premiações, não são funcionalidades abordadas pela plataforma. Para um usuário comum, qualquer tipo de estatística por ele desejada, necessitaria de um trabalho de pesquisa manual dentre aqueles arquivos disponibilizados.

Tendo em vista a importância da FEBRACE como uma das maiores feiras de ciências no cenário nacional, é extremamente relevante dados precisos de suas premiações anuais. Qual a região que mais se destacou em determinado ano? Qual estado tem o maior número de escolas diferentes premiadas? Qual a principal escola se destaca nas premiações da FEBRACE? De qual cidade/estado ela é? Quem foi o(a) principal orientador(a) dessa escola? Questionamentos como esses levariam a análises importantes sobre como é feito o trabalho científico em determinada região ou escola. Por exemplo, ao conversar com um orientador(a) que foi premiado diferentes vezes, pode-se entender qual tipo de abordagem esse professor(a)

realizada com seus alunos e como replica-la para obter melhores resultados em futuros projetos.

O projeto tem por objetivo destrinchar todos os PDFs disponibilizados pela plataforma, promovendo dados como: nome do projeto, escola, estado, cidade, categoria de premiação, integrantes, entre outros, no formato JSON e configurando uma estrutura mais harmoniosa de organização.

Em suma, os principais tópicos expostos neste artigo serão métodos e abordagens utilizadas para a modelagem de um database contendo diversas informações sobre a FEBRACE.

2 ESCOLHAS DE LINGUAGEM, BANCO DE DADOS E BIBLIOTECAS

2.1 PYTHON

Python é uma linguagem de programação de alto nível, fácil de aprender e se adaptar, com uma sintaxe clara e concisa é perfeita para scripts sequenciais lógicos. Além disso, possui uma ampla variedade de bibliotecas e módulos que facilitam a extração e manipulação de textos em PDF.

2.2 MONGO DB

O MongoDB é um software de banco de dados orientado a documentos livre, de código aberto e multiplataforma, escrito na linguagem C++. Classificado como um programa de banco de dados NoSQL, o MongoDB usa documentos semelhantes a JSON (GOOGLE,2023). O MongoDB permite que seja armazenado dados sem um esquema rígido, o que significa que pode ser adicionado novos campos aos documentos sem ter que alterar toda a estrutura do banco de dados, diferente de outras estruturas sql (FRANCISCATO,2023). É a escolha mais adequada para o projeto já que sua estrutura oferece uma interface de linha de comando intuitiva e fácil de usar, além de diversas ferramentas de gerenciamento de banco de dados disponíveis, sendo perfeito para tratamento de dados apenas de texto, como é o caso.

2.3 PYPDF2 E FITZ(PYMUPDF)

No python há grandes variedades de bibliotecas para extração de textos de pdfs, dentre essas bibliotecas estão o Fitz (PYMUPDF,2023) e o Pypdf2 (PYPDF2,2023). Para requisitos do projeto não há grande exigência quanto a funcionalidades dessas bibliotecas.

É necessário que a formatação do texto extraído seja idêntica a original, com os mesmos espaçamentos, acentuações, e quebras de linhas. Além disso, não há mais nenhum outro motivo para escolhas de diferentes bibliotecas.

3 ESTRUTURA NO MONGODB

3.1 CLUSTERS

No mongo é possível a criação de um cluster facilmente após realizar o login. O cluster é uma espécie de “servidor” onde ficará armazenado os databases do usuário. A versão gratuita de um cluster tem limite máximo de 512MB de dados, o que é muito mais do que o suficiente para armazenar os dados necessários no projeto.

3.2 DATABASE E COLLECTIONS

No mongo a hierarquia segue a seguinte ordem, um único cluster pode possuir vários databases, estes que podem possuir várias collections, onde cada collection possui vários documentos no formato JSON. No cluster atual do projeto a divisão se baseia em apenas um database com duas collection, uma collection que armazena todos os projetos extraídos do período, e outra para armazenar as escolas premiadas no período. Ambas as collections além de armazenar os respectivos nomes das escolas e dos projetos também apresentam dados importantes para cada.

Figura 1 - Estrutura de um documento da collection escolas

```
_id: ObjectId('64ede31a41675ee144d24ea5')
Escola: "Escola Agrícola de Jundiáí"
Cidade: "Macaíba"
Estado: "Rio Grande do Norte (RN)"
qntdpremiacao: 4
Premios: "2o. Lugar em Ciências Agrárias |Prêmio Destaque Unidades da Federação ..."
Orientadores: "Isaac Antunes Braga de Carvalho"
Anos: " 2020 4 vez(es)"
```

Figura 2 - Estrutura de um documento da collection de projetos

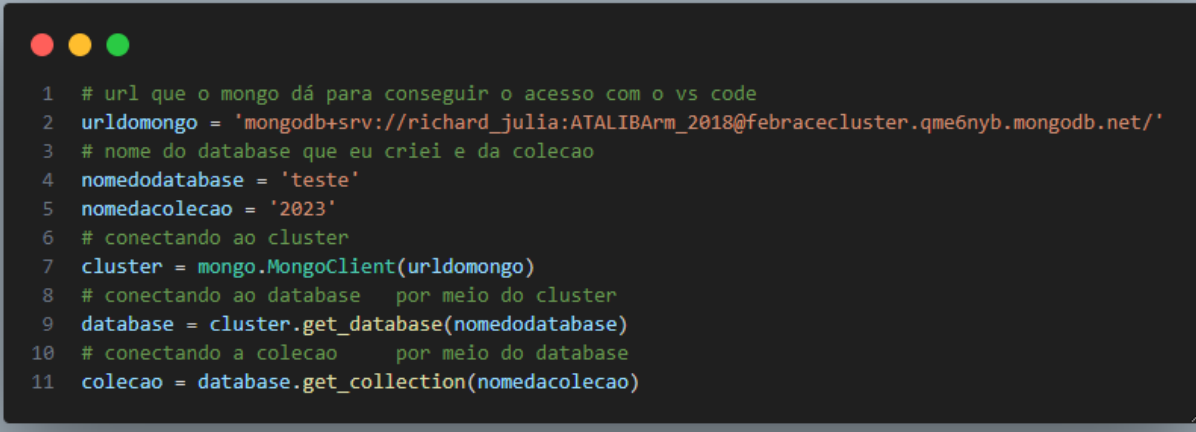
```
_id: ObjectId('64c15ef4ab4ac1247788522f')
Categoria_de_premiação: "Prêmio ABRIC de Incentivo à Ciência "
Nome_do_projeto: "Sistema de captação de água de poços artesianos sem uso de energia elé..."
Escola: "Escola Agrícola de Jundiáí"
Cidade: "Macaíba"
Estado: "Rio Grande do Norte (RN)"
Ano: "2020"
Orientador: "Isaac Antunes Braga de Carvalho"
Componentes: 2
Integrantes: "Alex Rosendo, Alzira Jeovania Borges de Oliveira, Isaac Antunes Braga ..."
```

3.3 SCRIPTS DO MONGO COM PYTHON

Como mostrado na figura logo acima, o mongo permite uma facilidade no momento da conexão via código python. A variável coleção na linha 11 é uma

referência para a coleção real criada no banco de dados “teste”. Essa referência pode ser usada para realizar operações CRUD (Criar, Ler, Atualizar, Deletar) nos documentos dentro dessa coleção

Figura 3 – Código conexão com mongo



```

1 # url que o mongo dá para conseguir o acesso com o vs code
2 urldomongo = 'mongodb+srv://richard_julia:ATALIBArm_2018@febracecluster.qme6nyb.mongodb.net/'
3 # nome do database que eu criei e da colecao
4 nomedodatabase = 'teste'
5 nomedacolecao = '2023'
6 # conectando ao cluster
7 cluster = mongo.MongoClient(urldomongo)
8 # conectando ao database por meio do cluster
9 database = cluster.get_database(nomedodatabase)
10 # conectando a colecao por meio do database
11 colecao = database.get_collection(nomedacolecao)

```

4 ESTRUTURA DOS PDF'S

4.1 INTRODUÇÃO A EXTRAÇÃO DE DADOS

É de suma importância para a organização e elaboração do projeto, a análise antecipada do conteúdo que será extraído no PDF. No decorrer do projeto algoritmos são criados para analisar centenas de páginas de PDFs. Com a diversidade de conteúdos presentes nesses arquivos, é necessário o entendimento de como os dados desejados estão alocados, qual a escalabilidade e confiabilidade deste conteúdo e etc, dessa maneira a criação dos algoritmos que irão estudar esses arquivos, pode ser mais precisa e coesa.

4.2 LISTA DE PREMIADOS

Na figura presente ao final deste tópico, está disponível a visualização da primeira página do PDF com a lista de premiados disponibilizados pela FEBRACE em 2023 (FEBRACE,2023). Diante disso, é notável a presença de 4 principais informações sobre o projeto premiado: prêmio, nome do projeto, cidade/estado, integrantes/orientador. Logo de início já é possível notar desconexões nas relações do mesmo projeto no arquivo de premiados e no de finalistas (mais detalhes no próximo tópico). Como dito anteriormente, é importante notar detalhes de padronização que existem na alocação dos dados, como por exemplo, o nome do projeto que sempre vem precedido de “ [PROJETO: “ e o prêmio que vem na linha posterior a linha com [PRÊMIO].

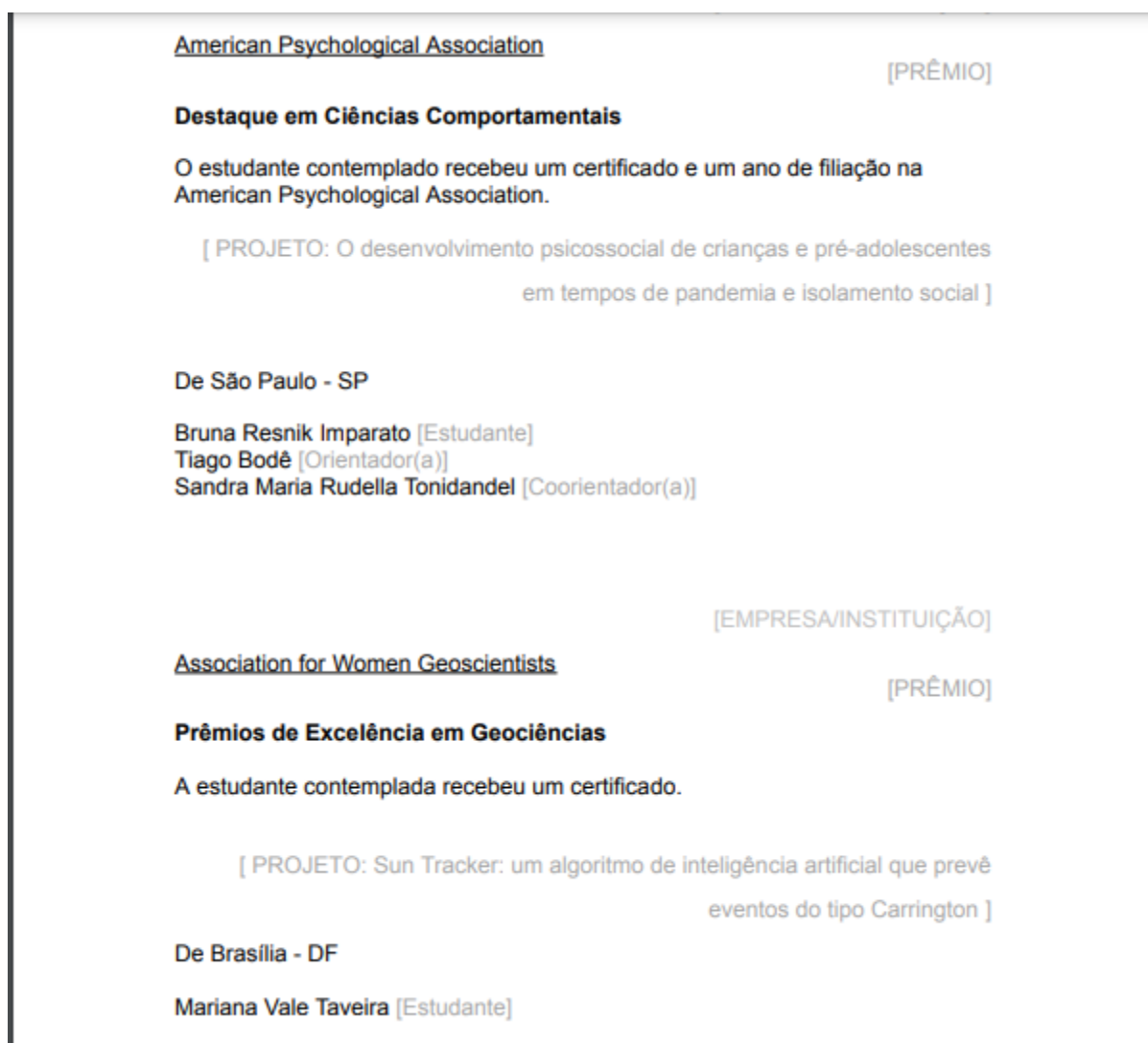


Figura 4

4.3 LISTA DE FINALISTAS

Nos textos em destaque na imagem abaixo, retirada do PDF de finalistas disponibilizado em 2023 (FEBRACE,2023), é possível notar o mesmo nome do projeto mostrado no tópico superior. Logo, pode-se observar 4 principais informações

expostas no arquivo para cada projeto: Nome do projeto. Integrantes, escola, cidade/estado. Também é possível notar que mesmo se tratando do mesmo projeto, há diferenças no mesmo dado apresentado no PDF de premiados e agora no PDF de finalistas, como por exemplo os integrantes destacados em negrito na *figura 5*, o qual no PDF de premiados, era visível apenas uma única aluna sem a presença do orientador.

Figura 5 – Exemplo lista de finalistas

SUN TRACKER: UM ALGORITMO DE INTELIGÊNCIA ARTIFICIAL QUE PREVÊ EVENTOS DO TIPO CARRINGTON (1983)
Mariana Vale Taveira, Roseno Gonçalves Lopes Filho, Leonardo Kuhn (Orientação) , Leandro Castelani (Coorientação)
 Colégio Ciman, Brasília, DF
 EXA - 105 Astronomia
Projeto da Feira: Mostra de Ciência e Tecnologia do Instituto Açaí-MCTIA

4.4 OBJETIVO COM OS PDFS

Visto os itens dos tópicos 4.2 e 4.3, pode-se organizar da seguinte forma

Tabela 1 – tabela de itens

DADOS FEBRACE	PDF PREMIADOS	PDF FINALISTAS
Nome do projeto	X	X
Localização(Cidade/estado)	X	X
Prêmio	X	X
Escola	X	X
Integrantes	X	X
Orientador	X	X

Assim pode-se concluir que, a relação entre os dois arquivos é incompleta e muitas vezes equivocada ao mostrar dados alterados para o mesmo item. A extração dos dados desses arquivos e organização no respectivo database irá proporcionar uma melhor visualização destes itens ao usuário, oferecendo um maior entendimento sobre a FEBRACE e suas premiações.

5 EXTRAÇÃO DE DADOS(COLLECTION PROJETOS)

5.1 ALGORITMO DE BUSCA DE DADOS

Quando a necessidade de desenvolver um programa ou rotina a ser executada pelo computador, precisamos deixar bem claro a sequência que deve ser seguida para atingir o resultado esperado. A esse encadeamento lógico na programação, chamamos de Lógica de Programação, e a descrição de como fazer, definimos como Algoritmos(BESSA,2023). O estudo feito no item 4. É fundamental no momento do desenvolvimento do código usado para a extração dos dados nos arquivos. Um

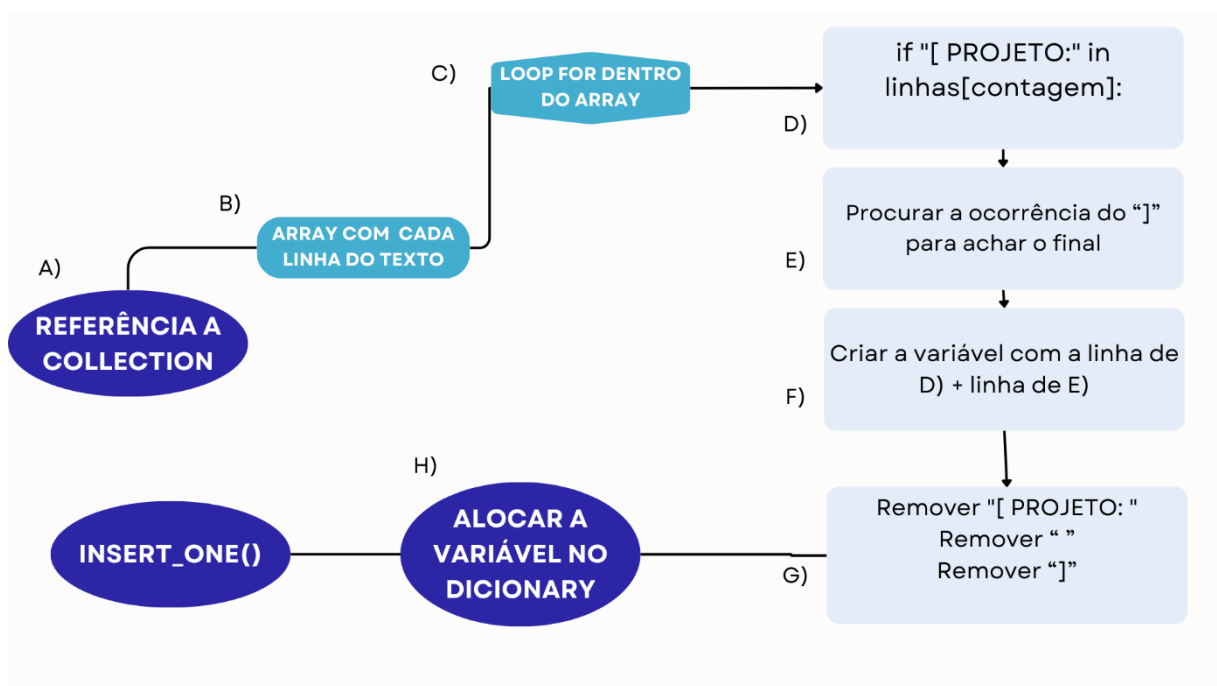
computador não pode facilmente identificar o que é um nome de projeto, ou o que é uma escola ou cidade/estado, um computador apenas segue algoritmos e lógicas de forma pragmática. É função do desenvolvedor a análise antecipada para a criação de uma estrutura lógica de algoritmos ideais para atingir o objetivo desejado.

Os algoritmos a seguir, expressos no formato de fluxogramas, seguem o mesmo padrão pois são um único arquivo dividido em mais de um pseudocódigo para facilitar a visualização. Começando com a criação da referência a collection citada no item 3.3 e na *figura 3*, e terminando com a criação de um dictionary em python para então o upload no mongo por meio do método do mongo: "insert_one".

5.2 NOME DO PROJETO

No arquivo de premiados 2023 mostrado na *figura 4*, é possível observar a formatação de escrita dos nomes dos projetos. Dessa forma pode-se desenvolver o algoritmo para análise de todo o conteúdo do .PDF em busca dos nomes.

Fluxograma 1 – Encontrando nome do projeto



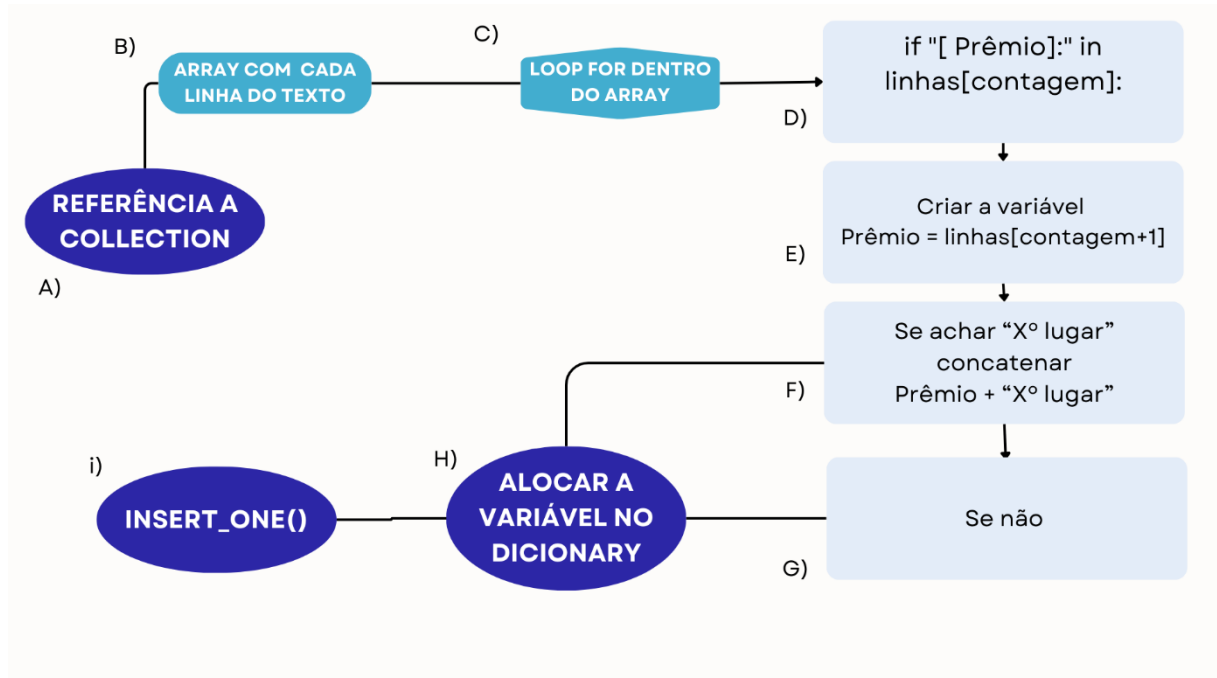
No item **F)** o código cria a variável concatenando as linhas onde forem encontrados os parâmetros do índice **D)** e do índice **E)**, e então a variável é alocada no banco de dados por meio do insert_one.

5.3 PRÊMIO

Os prêmios e os nomes dos projetos estão interligados, visto que os dados estão presentes no mesmo arquivo, dessa forma os métodos de array e loop

funcionam da mesma maneira, as diferenças estão basicamente nas estruturas condicionais.

Fluxograma 2 – Encontrando prêmios

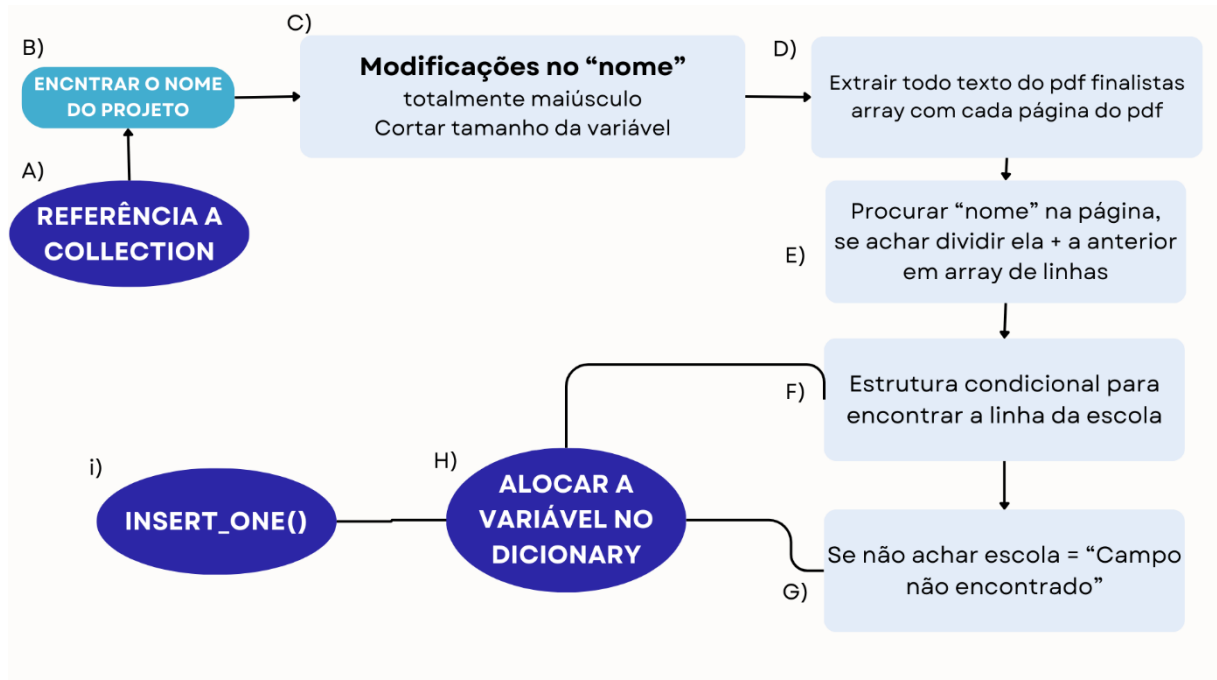


5.4 ESCOLA

O algoritmo responsável pela extração do item escola continua sendo parte do mesmo arquivo .PY visto nos itens acima. Entretanto, o item escola não está presente no mesmo arquivo que os itens anteriores, ele está presente apenas na lista de finalistas como já especificado na *tabela 1*. Assim, é preciso haver uma relação entre, achar o nome do projeto na lista de premiados, e em sequência pesquisar o mesmo na lista de finalistas para então encontrar a respectiva escola daquele projeto.

Do item **C)** sobre o fluxograma a seguir. Os nomes dos projetos na lista de finalistas estão totalmente em maiúsculos como mostrado em destaque na *figura 5*. Além disso, podem existir divergências na escrita do nome do projeto de uma lista para a outra, o “*corte*” do nome do projeto para fazer a pesquisa tem o fundamento de diminuir as chances de os projetos não serem encontrados na lista de finalistas, e por fim no mostrado no item **G)**.

Fluxograma 3 – Encontrando escolas

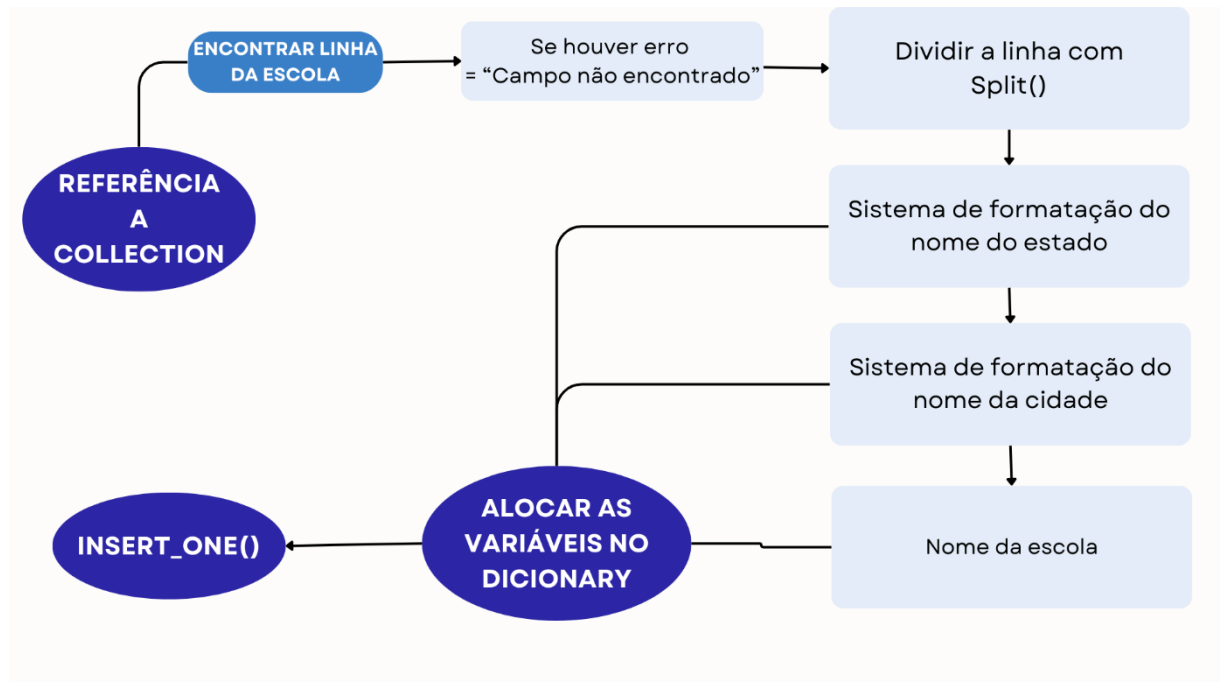


5.5 LOCALIZAÇÃO: CIDADE/ESTADO

Como visto no item 4.3, não existe uma única linha para a localização do projeto. Na verdade, a linha divide os campos : escola, cidade, estado, os quais geralmente estão separados por vírgulas, como pode ser atestado no exemplo da *figura 5*. Assim, a parte fundamental do algoritmo de encontrar a localização dos projetos, se baseia em dividir essa linha nas suas respectivas partes. Além disso, pode-se perceber que para cada linha o estado é informado apenas pela sua sigla, o que visualmente não se mostra adequado para formatos de tabela, e para futuros filtros ou objetos de pesquisa. Para modificar essa estrutura, o algoritmo passa por um sistema de detecção que analisa cada sigla para encontrar o nome do estado completo e então alocar os dados já divididos no database.

Vale ressaltar que, esse item mais os itens anteriores estão sendo compilados ao mesmo tempo, a área de alocar as variáveis no dictionary são comuns a todos os últimos itens, o método "insert_one" só é de fato compilado ao final do código onde todas as devidas variáveis já estão alocadas.

Fluxograma 4 – Localização



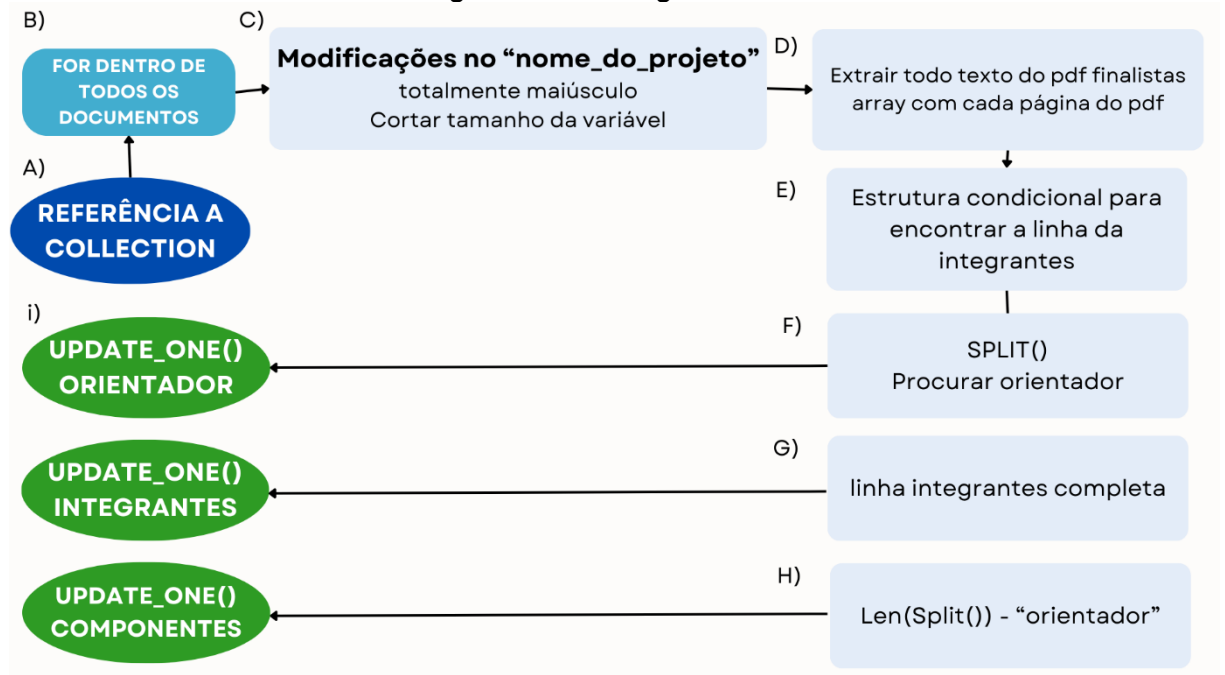
5.6 INTEGRANTES: ORIENTADOR, COMPONENTES

Os dados de integrantes e orientadores de cada projeto inicialmente não eram válidos para o objetivo da pesquisa, no entanto, são informações bem definidas e curiosas para uma visualização completa dos dados. O algoritmo de extração dos integrantes se passa em um arquivo posterior aos mostrados nos itens anteriores, os quais se passavam todos em conjuntos. Dessa forma são abordados agora não métodos de criação de documentos no banco de dados, mas sim formas de upload em documentos já existentes. Os algoritmos de extração dos integrantes são bastante semelhantes aos de escolas mostrada no item 5.4, visto que ambos estão bem próximos na mesma lista de finalista como mostrado no item 4.3.

A partir de agora os itens (representados em verde no fluxograma), não são mais algoritmos de CREATE como era o método `insert_one()`, e sim algoritmos de UPLOAD dentro dos documentos, utilizando o método do mongo: `"update_one()"`

Dessa forma os itens Orientador, componentes e integrantes mostrados na *figura 2* do item 3.2, são adicionados a todos os documentos do database, como mostrado de forma simplificada no fluxograma a seguir.

Fluxograma 5 – Integrantes



6 PESQUISA DE DADOS

6.1 COLLECTION ESCOLAS

Até então os documentos armazenados no database tinham enfoque principal para projetos premiados, as estatísticas e os itens se baseiam em informações para cada projeto: qual prêmio ele recebeu, qual o seu nome, em que ano foi e etc. Entretanto, existe a ideia de uma collection unicamente para informações das escolas, caso um usuário esteja curioso para entender a(s) participação(ões) de determinada escola na FEBRACE, a pesquisa pelos itens presentes na collection escolas pode passar uma noção informativa sobre a sua participação nas premiações da feira. Como já mostrado na *figura 1* a collection escolas possui 7 principais itens.

O mais interessante dentro dessa collection, é que todas as informações presentes nela, são adquiridas por meio das análises dos dados extraídos dos PDFs, assim comprovando como a análise da collection projetos pode ser usada de forma criativa para um estudo estatístico, e de variadas formas das premiações FEBRACE.

6.1.1 Pesquisa

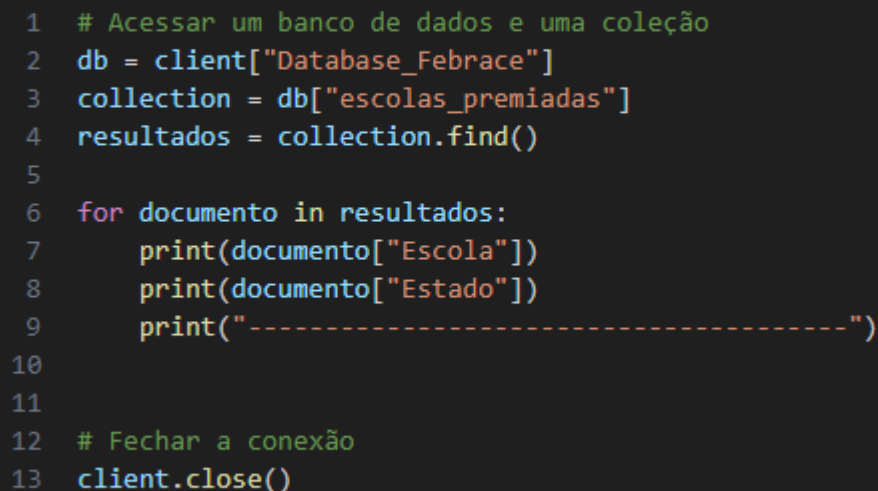
O mais interessante dentro dessa collection, é que todas as informações nela presentes, são adquiridas por meio das análises dos dados extraídos dos PDFs, assim

comprovando como o estudo da collection projetos pode ser usado de forma criativa para um exame estatístico, e de variadas formas das premiações FEBRACE. O mongo em adjunto com a linguagem python, permite sintaxes claras e simples de pesquisa e coleta de dados. Dessa forma, algoritmos são criados com o intuito de obter as informações mais relevantes para o estudo das escolas premiadas na FEBRACE.

6.1.2 Itens da collection

Na collection projetos estão cerca de 1300 documentos, todos com seus nomes de projeto, escolas e etc, porém fazendo um estudo dentro desses documentos, é visto que muitas escolas são repetidas para diferentes projetos, como a ideia é ter um database apenas para escolas, é sem sentido ter escolas repetidas, para isso é feita uma contagem dentro dos documentos, achando cerca de 420 escolas diferentes e adicionando cada uma na collection escolas. Após isso métodos de pesquisas são utilizadas para encontrar as informações dos itens restantes, contagem de quantas vezes a escola se repete para achar a quantidade de premiações; quais anos ela foi premiada, quais seus orientadores, sua localização e etc.

Figura 6 – Exemplo de busca dos campos escola e estado na collection escolas



```
1  # Acessar um banco de dados e uma coleção
2  db = client["Database_Febrace"]
3  collection = db["escolas_premiadas"]
4  resultados = collection.find()
5
6  for documento in resultados:
7      print(documento["Escola"])
8      print(documento["Estado"])
9      print("-----")
10
11
12  # Fechar a conexão
13  client.close()
```

7 CONCLUSÃO

Este trabalho apresentou técnicas, métodos, e tecnologias utilizadas para extrair dados das listas de premiados da FEBRACE no período 2018 – 2023, com o objetivo de auxiliar futuras pesquisas e estudos estatísticos sobre os participantes de suas premiações.

Os tópicos apresentados neste trabalho abstraem grande parte do código fonte utilizado na pesquisa. No entanto, os principais métodos e técnicas utilizadas não estão muito distantes do aprendizado na fração inicial do curso técnico em informática, conhecimentos básicos dentro da lógica de programação e algoritmos são meramente os itens necessários para a realização da pesquisa.

Com isso, pode-se usar esse trabalho como inspiração para futuros projetos com outras bancas além da FEBRACE, ou até incluir a continuidade dos arquivos anteriores ao período extraído. Embora tenha-se dedicado satisfatório tempo e esforço para a realização do projeto, o prazo disponível para a pesquisa e a coleta de dados foi restrito devido a diversas demandas acadêmicas e compromissos pessoais. Esta limitação de tempo influenciou a extensão das análises e experimentos realizados.

REFERÊNCIAS

FEBRACE. Premiados e finalistas 2023 - FEBRACE. 2023. Disponível em : <<https://febrace.org.br/premiados-e-finalistas/premiados-e-finalistas-2023/>>. Acesso em: 21/09/2023

FRANCISCATO. Qual a vantagem do mongodb sobre os outros bancos de dados? 2023. Disponível em : <<https://www.dio.me/articles/qual-a-vantagem-do-mongodb-sobre-os-outros-bancos-de-dados>>. Acesso em : 21/09/2023

GOOGLE. 2023. Disponível em : <<https://g.co/kgs/iGSrW4>>. Acesso em : 21/09/2023

PYMUPDF. Module fitz - PyMuPDF 1.23.3 documentation. 2016. Disponível em : <<https://pymupdf.readthedocs.io/en/latest/module.html>>. Acessado em : 21/09/2023

PYPDF2. PyPDF2. 2022. Disponível em : <<https://pypi.org/project/PyPDF2/>>. Acesso em : 21/09/2023

BESSA. Lógica de programação e algoritmos - ALURA. 2023. Disponível em: <<https://www.alura.com.br/artigos/algoritmos-e-logica-de-programacao#introducao>>. Acesso em: 21/09/2023

AGRADECIMENTOS

A minha dupla Julia P. pelo trabalho realizado em conjunto.

A os meus familiares por todo o apoio do começo ao final do curso.