



FACULDADE PROJEÇÃO

Engenharia de Software

Nádio Dib Fernandes Pontes

**Análise e correlação de dados de boletins do ONS para predição de
incidentes de sobrecarga energética em UHEs de acordo com as
resoluções normativas da ANEEL através de frameworks de Data Mining e
Web Scraping**

Brasília/DF, 17 de Maio de 2024



FACULDADE PROJEÇÃO

Engenharia de Software

Análise e correlação de dados de boletins do ONS para predição de incidentes de sobrecarga energética em UHEs de acordo com as resoluções normativas da ANEEL através de frameworks de Data Mining e Web Scraping

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Software da Faculdade Projeção, como requisito complementar e obrigatório à obtenção do título de Bacharel em Engenharia de Software.

Brasília, 17 de Maio de 2024.

Banca Examinadora:

Professor Orientador: Fernando Feliu González

Professor(a):

Professor(a):

Rosa Maria Diekn de Queiroz, Mestre
Coordenadora do curso de Sistemas de Informação

AGRADECIMENTOS

Primeiramente, à minha família, Josiane, Alexandre e Luana, no qual fizeram parte dessa longa jornada durante minha graduação. Também ao meu inspirador, prof. Kauê, no qual me motivou e inspirou a dar início ao curso de Engenharia de Software e continuar resiliente, independente das adversidades nos quais enfrentaria. Meus agradecimentos à minha turma do curso de Engenharia de Software, colegas dos cursos de Sistemas de Informação e Análise e Desenvolvimento de Sistemas, além de meus colegas mais próximos de turma William, Ricardo, Isaac e Yuri.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS	6
LISTA DE FIGURAS	7
LISTA DE TABELAS	8
RESUMO.....	10
ABSTRACT.....	11
CAPÍTULO I.....	12
VISÃO INICIAL	12
1. Introdução.....	12
2. Contextualização.....	12
3. Problemática.....	13
4. Solução Proposta.....	13
5. Objetivo Geral.....	14
5.1. Objetivos Específicos	14
6. Estrutura do Projeto.....	15
CAPÍTULO II.....	16
METODOLOGIA APLICADA.....	16
1. Metodologia e Modelo de Desenvolvimento	16
2. Técnicas de Levantamento de Requisitos	16
2.1 Brainstorming.....	16
2.2 Observação Direta.....	17
3. Principais tecnologias utilizadas	18
3.1 Selenium WebDriver.....	18
3.2 Python 3.....	19
3.3 MariaDB Server	20
3.4 FastAPI	21
3.5 Uvicorn.....	23
3.6 Prophet	23
3.7 NumPy	24
3.8 Splinter.....	25
4. Análise de Sistemas Existentes.....	25
CAPÍTULO III.....	27
VISÃO DO SISTEMA.....	27

1. Riscos do Projeto.....	27
2. Restrições do Projeto	27
3. Regras do Negócio.....	28
4. Requisitos Funcionais	29
5. Requisitos Não Funcionais	30
6. Mensagens do Sistema.....	31
7. Lista de Casos de Uso	31
8. Lista de Atores	33
9. Diagrama Geral de Caso de Uso	34
10. Diagrama de Classe	35
11. Diagrama de Sequência.....	36
CAPÍTULO IV.....	37
ESPECIFICAÇÃO DE CASO DE USO	37
UC001 – Acionar Bot de extração.....	37
UC002 – Extrair dados do ONS	39
UC003 – Processar dados extraídos via frameworks <i>Data Mining</i>	41
UC004 – Atualizar registros na base de dados	43
UC005 – Excluir arquivos temporários	45
UC006 – Consultar predição via endpoint por período	46
UC007 – Extrair dados dos boletins publicados dos subsistemas	48
UC008 – Realizar predição estocástica das séries temporais via framework Prophet	49
CAPÍTULO V.....	51
MODELAGEM DE DADOS.....	51
1. Diagrama de Entidade Relacional.....	51
2. Diagrama de Entidade Relacional.....	51
3.1 Dicionário de Dados	52
3.1.1 Tabela [Nome da Tabela]	52
1.1 Campos da Tela	53
1.2 Comandos da Tela	53
CONCLUSÃO	54
REFERÊNCIAS.....	55

LISTA DE ABREVIATURAS E SIGLAS

ANEEL	Agência Nacional de Energia Elétrica
SIN	Sistema Integrado Nacional
ONS	Operador Nacional do Sistema Elétrico
UHE	Unidade Hidrelétrica
ER	Entidade Relacional
CAG	Controle Automático de Geração
W3C	World Wide Web Consortium
IDE	Integrated Development Environment
ASGI	Asynchronous Server Gateway Interface
WSGI	Web Server Gateway Interface
SisBIN	Sistema de Boletins Integrado Nacional
JSON	JavaScript Object Notation
API	Application Platform Interface
REST	Representational State Transfer
RESTful	Segue uma implementação mais completa e restrita dos princípios REST.
Bot	Robô automatizado através de operações computacionais.
Endpoint	Endereço eletrônico utilizado por serviços através de uma API.
AWS Open Data	Conjunto de dados de alto valor otimizados para a nuvem e disponíveis para o público no qual é hospedado pela empresa Amazon.

LISTA DE FIGURAS

Figura 1: arquitetura simplificada do Selenium WebDriver e suas iterações com os principais navegadores da internet.	19
Figura 2: breve abordagem das principais vantagens de escolher a linguagem Python.	20
Figura 3: principais características da MariaDB.	21
Figura 4: comparação do FastAPI com os principais frameworks de APIs do mercado, onde enfatiza a melhor escolha sendo a intermediária.	22
Figura 5: Uvicorn promove um ganho considerável na performance do servidor ASGI.....	23
Figura 6: Prophet auxilia nas previsões práticas através da criação de modelos de regressão modular para análise em séries temporais.	24
Figura 7: uma das maiores ferramentas de computação numérica para cálculos vetoriais n-dimensionais de modo rápido e versátil.	24
Figura 8: Diagrama de Caso de Uso do SisBIN para representação das operações diretas à API e afins.....	34
Figura 9: Diagrama de Classes.	35
Figura 10: Diagrama de Sequência.....	36

LISTA DE TABELAS

Tabela 1: Riscos do Projeto.....	27
Tabela 2: Restrições do Projeto.....	27
Tabela 3: Regras de Negócio.	28
Tabela 4: Requisitos Funcionais.....	29
Tabela 5: Requisitos Não Funcionais.	30
Tabela 6: Mensagens do Sistema.	31
Tabela 7: Lista de Caso de Uso.....	31
Tabela 8: Lista de Atores.....	33
Tabela 9: fluxo principal referente ao caso de uso UC001 - Acionar Bot de extração.....	37
Tabela 10: fluxo alternativo referente ao caso de uso UC001 - Acionar Bot de extração.....	38
Tabela 11: fluxo de exceção referente ao caso de uso UC001 - Acionar Bot de extração.....	38
Tabela 12: fluxo principal referente ao caso de uso UC002 – Extrair dados do ONS.	39
Tabela 13: fluxo de exceção referente ao caso de uso UC002 – Extrair dados do ONS.	40
Tabela 14: fluxo principal referente ao caso de uso UC003 – Processar dados extraídos via frameworks Data Mining.	41
Tabela 15: fluxo de exceção referente ao caso de uso UC003 – Processar dados extraídos via frameworks Data Mining.	42
Tabela 16: fluxo principal referente ao caso de uso UC004 - Atualizar registros na base de dados.....	43
Tabela 17: fluxo de exceção referente ao caso de uso UC004 - Atualizar registros na base de dados.....	43

Tabela 18: fluxo principal referente ao caso de uso UC005 – Excluir arquivos temporários.	45
Tabela 19: fluxo principal referente ao caso de uso UC006 – Consultar previsão via endpoint por período.....	47
Tabela 20: fluxo principal referente ao caso de uso UC007 – Extrair dados dos boletins publicados dos subsistemas.	48
Tabela 21: fluxo principal referente ao caso de uso UC008 – Realizar previsão estocástica das séries temporais via framework Prophet.....	49

RESUMO

Este Trabalho de Conclusão de Curso, tem como objetivo explicitar a necessidade da coleta de dados dos boletins gerados pelo Operador Nacional do Sistema Elétrico (ONS), para fins de predição de incidentes de sobrecarga energética em Unidades Hidrelétricas (UHEs) de acordo com as regulações normativas da Agência Nacional de Energia Elétrica (ANEEL) através de tecnologias de *Data Mining* para mineração de dados extraídos por robôs programáveis, nos quais utilizam *frameworks* de *Web Scraping*. Os possíveis incidentes nas UHEs serão detectáveis por análises correlativas e classificatórias, além de serem norteados pelas normas da ANEEL quanto a sua detecção e prevenção.

Palavras-Chave: ONS; ANEEL; Data Mining; Web Scraping; Web Crawling; prevenção contra incidentes em UHEs; análise correlativa de dados em boletins do ONS; análise classificatória e preditiva de dados em boletins do ONS.

ABSTRACT

The purpose of this Completion of Course Work is, to explicit how important is to collect data from National Electrical Operator System (NOS), to provide an incident foresight to Hydro Power Plants (HPPs) against overwhelm energy overload according to standard regulations from National Electrical Energy Agency (NEEA) using technologies and frameworks of Data Mining and Web Scraping. Therefore, possible incidents on HPPs could be found among data correlation, data classification and prediction analysis when applying NEEA standard regulations to detect and mitigate them.

Keywords: NOS; NEEA; Data Mining; Web Scraping; Web Crawling; incident foresight on HPPs; data correlation analysis on NOS bulletins; data classification and prediction analysis on NOS bulletins.

CAPÍTULO I

VISÃO INICIAL

1. Introdução

O Operador Nacional do Sistema Elétrico (ONS) [1] é o órgão responsável pela coordenação e controle da operação das instalações de geração e transmissão de energia elétrica no Sistema Interligado Nacional (SIN) e pelo planejamento da operação dos sistemas isolados do país, sob a fiscalização e regulação da Agência de Energia Elétrica (ANEEL).

Além de ser um órgão que desempenha o papel de monitoração do sistema elétrico brasileiro, também disponibiliza um acervo de boletins, nos quais podem ser consultados via endereço eletrônico, em <http://sdro.ons.org.br/SDRO/DIARIO/index.htm>. Tais boletins são publicados diariamente, desde 01 de janeiro de 2000¹, onde o primeiro registro foi incluído.

Um dos principais propósitos desses boletins é de prover informações quanto ao balanço de energia, produção de energia, carga e demanda, integração de equipamentos e ocorrências no SIN e hidrologia.

Todavia, quando interligadas essas informações podem ser relacionadas para outros fins, como por exemplo, o próprio tema deste trabalho, no qual tem como objetivo correlacionar esses dados que estão isolados em diversos boletins diários e assim possibilitar um modelo de predição de dados unificado para prevenção de possíveis incidentes em UHEs registradas no Brasil.

2. Contextualização

Atualmente, o SIN, que representa o sistema de energia elétrico nacional, nos quais fazem parte os subsistemas Sudeste, Centro-Oeste, Sul, Nordeste e

¹ O ONS fornece acesso a todos os boletins gerados desde 01 de janeiro de 2013, se for acessado pelo endereço eletrônico, em <http://sdro.ons.org.br/SDRO/DIARIO/index.htm>. Entretanto, após o advento do Open Data ONS, agora é possível ter acesso ao acervo de publicações desde 01 de janeiro de 2000 para formato de dados em XLSX e CSV, nos quais são mantidos pela AWS Open Data.

Norte, consta com demandas de carga energética diária para atender os requisitos de abastecimento elétrico nessas regiões.

Entretanto, em certos períodos do ano, o nosso país, Brasil, ocasionalmente recebe tantos eventos nacionais, regionais ou mesmo internacionais que, diretamente, necessitam do fornecimento elétrico para atender a demanda das residências, indústrias, comércio e outros envolvidos no contexto.

Sendo assim, apesar do SIN ser, em sua grande maioria, automatizado quanto a entrega de cargas nas regiões para atendimento da demanda energética, os procedimentos de monitoração ainda são manuais e contam com o suporte do ONS e normas regulatórias da ANEEL para que sobrecargas desnecessárias não sejam acionadas incorretamente para essas regiões, segundo o Planejamento Anual da Operação Energética para margens de 5 anos [1].

3. Problemática

O principal problema noticiado é a necessidade da unificação destas informações, referente aos boletins diários fornecidos pelo ONS, para análises preditivas de longos períodos, como por exemplo previsões anuais, ao invés de serem realizadas através de observações diárias.

Através deste panorama, será possível contornar alguns tópicos e possibilitar outros benefícios, utilizando tecnologias e mecanismos presentes na atualidade para fornecer ainda mais vantagens para este contexto, por exemplo como a criação de modelos de aprendizagem de máquina nos quais performem tarefas de predição contra eventos e incidentes iminentes em UHEs, baseados nos dados informados pelos boletins diários do ONS.

4. Solução Proposta

Devido a necessidade de coletar dados dos boletins gerados pelo ONS e elaborar artefatos como a criação de relatórios específicos com informações contra possíveis incidentes em UHEs, como por exemplo quanto o levantamento de informações geradas por estes boletins para margens de período, mais

amplo, como mensal, semestral ou anual, este problema afeta diretamente os órgãos ONS e ANEEL.

Contudo, o desenvolvimento deste sistema tem como benefício providenciar a criação de uma API pública através de arquitetura RESTful para consulta destes dados relatados anteriormente, nos quais serão obtidos por robôs de extração de dados, que utilizam frameworks para este fim, de modo a atender estas expectativas do projeto. Além que será providenciado a geração de relatório com margens de previsão de possíveis incidentes no SIN via modelagens de correlação de dados através desta mesma API em formato de dados via JSON.

5. Objetivo Geral

O objetivo deste documento é contribuir para o aprimoramento e aperfeiçoamento da qualidade das informações disponibilizadas além de coletar, analisar e definir necessidades e características gerais do projeto referente à análise preditiva de incidentes em UHEs via metodologias de correlação de dados por frameworks de mineração de dados e robôs para extração de dados.

5.1. Objetivos Específicos

Dado o contexto da análise preditiva de incidentes nos quais envolvem as UHEs, este trabalho tem como visão além de atender as expectativas citadas no objetivo geral, também os seguintes objetivos específicos abaixo:

- Beneficiar a facilitação na coleta de dados para a geração de modelos de aprendizagem de máquina através de demais frameworks de *Machine Learning*; e
- Beneficiar o uso de tecnologias que utilizam frameworks de mineração de dados para criação de modelos preditivos no contexto de predição de incidentes mais complexos de identificar, bem como corroborar no incremento de análises estatísticas mais elaboradas que poderão ser relacionadas com os dados numéricos e classificatórios dos boletins diários do ONS.

6. Estrutura do Projeto

Este projeto de conclusão de curso foi dividido em capítulos nos quais auxiliarão na leitura e compreensão de seus tópicos mais específicos, onde se tentou apresentar de forma mais aprofundada os itens referentes ao tema principal envolvendo este sistema e suas ramificações.

- Capítulo I: é referente à visão inicial e demais relações do projeto em questão, no qual especificam-se tópicos relacionados ao contexto do projeto, motivações sobre os pontos de melhoria, análises gerais e específicas e a atual estrutura deste documento;
- Capítulo II: é referente à metodologia aplicada no desenvolvimento deste trabalho, no qual enfatiza alguns padrões e tópicos nos quais proporcionaram a evolução e progresso de padrões técnicos aplicados à este projeto, bem como enfoque nas tecnologias, frameworks e demais informações de análises e prazos estipulados que foram aplicados neste trabalho;
- Capítulo III: é referente à visão de todo o sistema desenvolvido, aplicando o conhecimento técnico e profissional no qual foi compartilhado durante o curso de bacharelado de Engenharia de Software de acordo com as melhores práticas e metodologias proporcionadas na formação pela instituição, neste capítulo também é possível obter detalhes sobre o sistema em questão referente a este trabalho;
- Capítulo IV: é referente as especificações técnicas de Casos de Usos proporcionados para o devido andamento de passos essenciais na criação de funcionalidades deste sistema em questão;
- Capítulo V: é referente aos padrões de modelagem de dados utilizados na concepção de bases de dados preferível para persistência das informações geradas pelo sistema, de modo a promover melhor compreensão de sua estruturação e significado.
- Enfim temos os demais componentes deste trabalho como: Conclusão e Referências.

CAPÍTULO II

METODOLOGIA APLICADA

1. Metodologia e Modelo de Desenvolvimento

A metodologia empregada na implantação deste projeto, para Análise e correlação de dados de boletins do ONS para predição de incidentes de sobrecarga energética em UHEs, é o modelo de desenvolvimento em cascata ou *waterfall*. Esta metodologia tem como principal a ênfase de conceitos e fundamentos sólidos de todo este sistema na qual será desenvolvido.

Segundo Hickey e Davis [2], afirmam que é imprescindível que, para o desenvolvimento de software, o conhecimento para a evolução do projeto é fundamental tanto para o embasamento dos principais requerimentos levantados durante a técnicas de elicitação quanto para a construção de documentos de análise, prototipação e demais métodos para obtenção de requisitos utilizados durante o refinamento dos principais requisitos do sistema, nos quais constarão neste Projeto de Conclusão de Curso.

O modelo de desenvolvimento *waterfall* tem como principais vantagens [3] a criação dos requerimentos nas fases iniciais, onde são cruciais para a criação do escopo, demais tarefas e design deste projeto; melhoria na utilização de recursos; melhor compreensão durante o desenvolvimento deste projeto; e constante feedback durante todo os processos, quanto a seu andamento.

2. Técnicas de Levantamento de Requisitos

2.1 Brainstorming

Esta técnica é utilizada para realização de discussões informais com o intuito de expressar o livre arbítrio de novas ideias e concepções para o desenvolvimento do sistema.

O Brainstorming [4] funciona em duas fases, sendo estas: a fase de geração, na qual as ideias são coletadas sem julgamentos ou critérios; e a fase de evolução, onde através das ideias coletadas também serão discutidas.

O principal intuito para a utilização desta técnica neste projeto vem através de aplicar conceitos de inovação e necessidade de unificar conceitos já abstraídos o mercado para o contexto desta problemática.

O objetivo principal da aplicação desta técnica no sistema foi devido a mescla de experiências prévias tanto com o desenvolvimento de robôs de extração de dados de conteúdo de cunho científico como IEEE, no qual se objetivava a geração de relatórios estatísticos para criação de publicações e apresentações acadêmicas ao departamento de Engenharia Elétrica da Universidade de Brasília, no ano de 2020, no período de agosto até dezembro.

Além de experiências a respeito da criação de robôs nos quais utilizavam frameworks que promovia a extração de informações específicas, também obtive outras experiências no contexto da Engenharia Elétrica nos anos de 2018 a 2019, onde meu antigo professor que ministrava as disciplinas de gerenciamento elétrico, citou a possibilidade de avaliar os boletins do ONS de forma automatizada ao invés de manual.

Sendo assim, esses foram os principais quesitos que possibilitaram a aplicação desta técnica para o avanço tanto de uma Prova de Conceito quanto implementações através de código-fonte e, conseqüentemente, progredir no desenvolvimento deste sistema objetivado neste documento.

2.2 Observação Direta

Esta técnica etnográfica é utilizada através da observação e envolvimento direto do observador na sociedade e geralmente é utilizada em conjunto com outras técnicas a fim de coletar requisitos implícitos que a técnica de entrevista não consegue revelar.

A Observação Direta [4] é feita através de dois processos: ativo, no qual a abordagem de questionamento é feita ao usuário durante o processo de observação; e passiva, onde o observador não interage com o usuário durante a observação de sua rotina.

O principal intuito para a utilização desta técnica neste projeto é devido a não existência de projetos ou iniciativas similares em outros projetos acadêmicos, além de prover insumo para o desenvolvimento de outros mecanismos de proteção de incidentes em UHEs.

A técnica de Observação Direta está diretamente interligada a técnica evidenciada anteriormente, Brainstorming, pois mesmo com os protocolos de segurança e demais funções regulatórias de órgãos nacionais como a ANEEL, ainda existe as possibilidades de falha humana no processo de monitoração, que é feito em algumas etapas pelo próprio ONS.

Logo, devido a esta objeção, análises e relatos de profissionais que atuam neste contexto sobre a influência de possíveis incidentes em UHEs, foi evidente a necessidade de um providenciar mecanismo que fornecesse uma melhoria neste processo de segurança.

Por mais calculistas e precavidos contra esses incidentes, ainda existe a possibilidade iminente de acidentes ocorrê-los devido a operação humana em etapas do processo de geração energética dentro da matriz do SIN através das UHEs.

3. Principais tecnologias utilizadas

O sistema será testado em um ambiente computacional com as seguintes características:

- Processador AMD Ryzen 7 5700G 4.30 GHz – 16 processadores lógicos;
- Placa de gráfica NVIDIA GeForce RTX 3060 – 12 GB de memória dedicada e 16 GB de memória compartilhada;
- Memória virtual (RAM) de 32 GB DDR4 2666MHz;
- Disco rígido HS-SSD-E1000 512G;
- Conexão via fibra óptica de 500MB/s para driver de rede Realtek PCIe 2.5GbE Family Controller; e
- Sistemas operacionais: Windows 11 Pro 23H2 e Linux Ubuntu 20.04 LTS.

3.1 Selenium WebDriver

O Selenium WebDriver [5] opera o navegador ou *browser* nativamente, bem como o usuário, seja localmente ou ambiente virtual remoto através de um servidor Selenium, esse framework permite a automação de processos dentro do navegador.

Selenium WebDriver Architecture

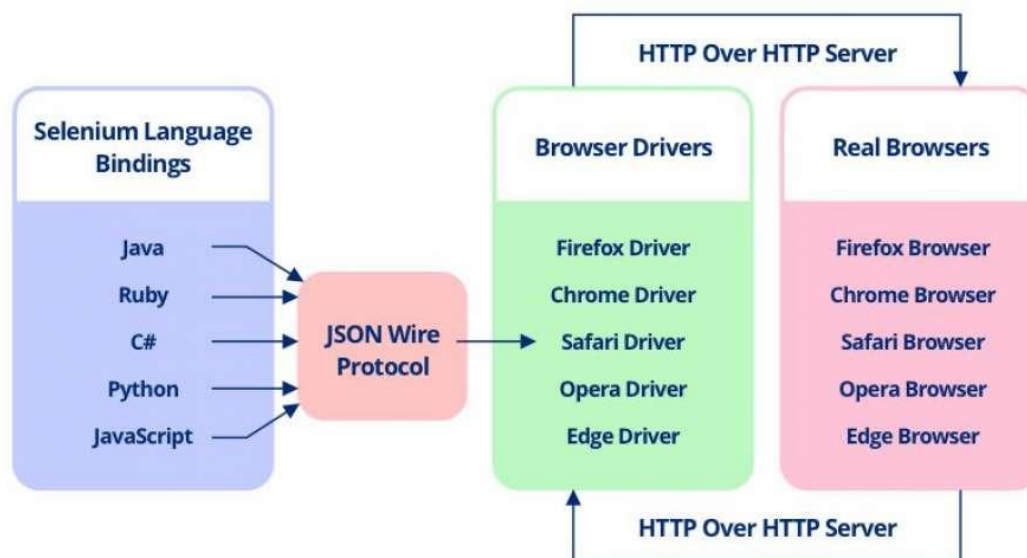


Figura 1: arquitetura simplificada do Selenium WebDriver e suas iterações com os principais navegadores da internet.

Este framework refere-se tanto a interligação com a linguagem de programação quanto com a implementação via controle de código partes individuais do *browser*. Popularmente é referido apenas com o termo WebDriver.

O Selenium WebDriver é uma recomendação da W3C [6]:

- O WebDriver foi designado para atuar como uma interface simples e concisa para programação de processos automatizados no navegador;
- O WebDriver fornece uma API compactada orientada à objeto; e
- Permite o controle do navegador de forma eficiente.

3.2 Python 3

O Python [7] é uma linguagem de programação de fácil aprendizagem e ao mesmo tempo poderosa. Ela possui um alto nível de controle sobre as principais estruturas de dados e de forma simplificada atua de modo eficiente na abordagem da programação orientada à objetos. Sendo assim, de forma elegante e dinâmica na sua escrita, junto com seu interpretador de código integrado nativo, promove uma linguagem de programação ideal para a criação

de scripts em aplicações que demandam rápido desenvolvimento em diversas áreas e plataformas.

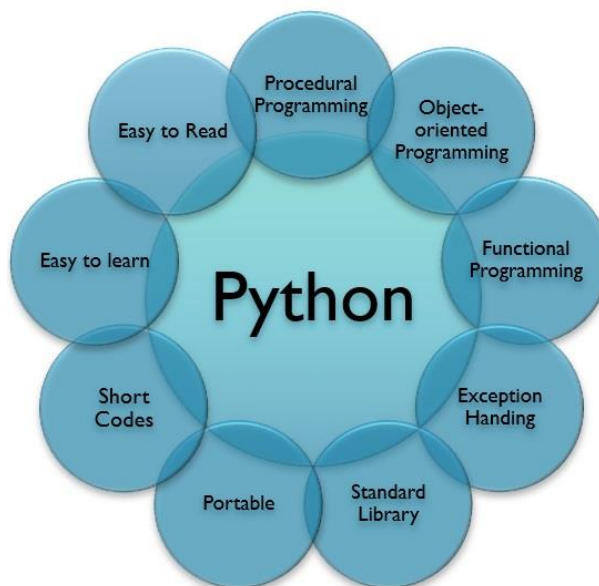


Figura 2: breve abordagem das principais vantagens de escolher a linguagem Python.

O interpretador do Python e sua diversa gama de bibliotecas padronizadas são facilmente disponíveis para obtenção tanto em código fonte original quanto de seus binários para todas as principais plataformas no próprio site da linguagem [7], além de promover o seu uso para diversos propósitos.

Além disso, o interpretador Python permite a extensão de funções e tipos de dados implementados em C ou C++ [8] [9] (ou qualquer outra linguagem de programação oriunda do C nativo). Sendo assim, o Python é comumente escolhido para ser um extensor de linguagens para customização de aplicações.

3.3 MariaDB Server

A MariaDB Server [10] é um dos mais populares servidores para banco de dados do mundo. Originalmente foi desenvolvido pelos mesmos criadores do MySQL e se mantém *open source*.

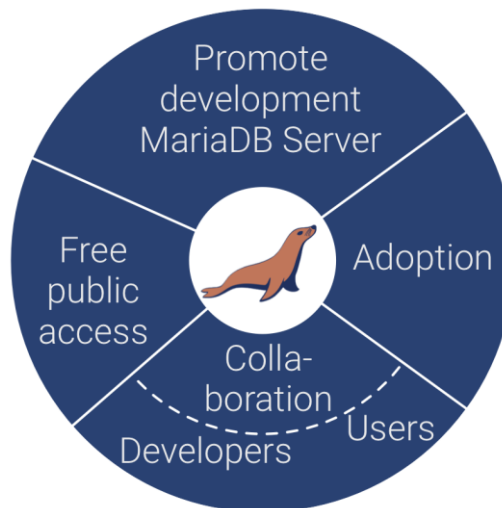


Figura 3: principais características da MariaDB.

MariaDB converte informações e dados de diferentes estruturas em uma grande variedade de aplicações, desde bancos a websites. Foi designada para atuar de forma otimizada e melhorada de modo a substituir o MySQL, onde atuar de maneira rápida, escalável e robusta e preserva um rico ecossistema para persistência de dados dentro das principais *engines*, plugins e demais ferramentas versáteis de banco de dados do mercado e diversas aplicações para diferentes casos de usos.

Além disso, foi desenvolvida como um projeto de software de código aberto e como um banco de dados relacional onde fornece suporte uma interface de interpretação do SQL para acesso aos dados. As últimas versões do MariaDB incluem suporte ao GIS e funcionalidades JSON.

3.4 FastAPI

O FastAPI [11] é um moderno framework web de alta performance para construção de APIs utilizando o Python 3.6+ baseado nos principais padrões do Python e recomendações.



Figura 4: comparação do FastAPI com os principais frameworks de APIs do mercado, onde enfatiza a melhor escolha sendo a intermediária.

Fornece as principais funcionalidades chave como:

- Rapidez: alta performance, quando pareado com aplicações em NodeJS e Go;
- Produtividade: aumento na velocidade de produção de novas funcionalidades em 200% a 300%, segundo estimativas em testes com o time de desenvolvimento ao realizar a criação de aplicações em ambiente de produção;
- Menos bugs: redução de 40% da inclusão de bugs acidentalmente por erro humano, segundo estimativas em testes com o time de desenvolvimento ao realizar a criação de aplicações em ambiente de produção;
- Intuição: fornece amigável integração com IDEs ao prover documentação de funcionalidades internas, complementação de código dinâmica e menor tempo durante as depurações de programa;
- Facilidade: foi desenhada para ser fácil de usar e aprender, requerindo menos tempo para ler documentações;
- Minimalista: redução da criação de código duplicado, além de prover várias funcionalidades para cada declaração de parâmetro, resultando em menos bugs;
- Robusta: obtenha códigos mais rápidos em ambiente de produção com a documentação interativa;

- Baseado nos melhores padrões: baseado e totalmente compatível com os padrões e definições das APIs, como OpenAPI [12] (previamente conhecido como Swagger) e JSON Schema [13].

3.5 Uvicorn

O Uvicorn [14] é um otimizador de performance para implementação de servidores ASGI utilizando bibliotecas Python uvloop [15] e httptools [16].



Figura 5: Uvicorn promove um ganho considerável na performance do servidor ASGI.

Devido a necessidade de implementação funcionalidades de processamento assíncrono em aplicações Python, o Uvicorn veio para cobrir essa demanda. Com a introdução do ASGI as aplicações permitiram que frameworks com essa necessidade permitissem esse tipo de processamento.

O ASGI deve fornecer suporte no ecossistema Python de aplicações web e seus demais frameworks onde são altamente competitivos contra os de NodeJS e Go em termos de alto tráfego de informações nos seus limites de E/S (Entrada/Saída). Além disso, promove o suporte para HTTP/2 e WebSockets, nos quais podem ser manipulados através de servidores WSGI.

O Uvicorn também fornece suporte para HTTP/1.1 e WebSockets, onde a implementação de suporte HTTP/2 ainda está sendo planejada.

3.6 Prophet

O Prophet [17] utiliza de processos estocásticos para criar modelos estatísticos de previsão de dados de séries temporais com base em um modelo aditivo onde tendências não lineares são ajustadas quanto a sua sazonalidade

anual, semanal e diária. Este framework foi desenvolvido e lançado pela equipe Core Data Science do Facebook e está disponível para download nas plataformas CRAN e PyPI.



Figura 6: Prophet auxilia nas previsões práticas através da criação de modelos de regressão modular para análise em séries temporais.

O procedimento de previsão é uma tarefa comum da ciência de dados que ajuda as organizações no planejamento de capacidade, definição de metas e detecção de anomalias. Apesar da sua importância, existem sérios desafios associados à produção de previsões confiáveis e de alta qualidade.

Para enfrentar esses desafios, a equipe envolvida no desenvolvimento do framework propôs um modelo de regressão modular com parâmetros interpretáveis que podem ser ajustados com conhecimento de domínio sobre as séries temporais. Além disso, análises de desempenho para comparar e avaliar procedimentos de previsão e sinalizar automaticamente as previsões para revisão e ajuste manual. Ferramentas estas que auxiliam os especialistas em analisar conhecimentos de forma mais eficaz e assim permitir previsões mais práticas e confiáveis de séries temporais [18].

3.7 NumPy

O NumPy [19] é uma biblioteca de código aberto que possibilita a computação numérica usando a linguagem de programação Python.



Figura 7: uma das maiores ferramentas de computação numérica para cálculos vetoriais n-dimensionais de modo rápido e versátil.

Criado em 2005, ele se baseou nos trabalhos anteriores das bibliotecas Numeric e Numarray. O NumPy sempre será um software 100% gratuito e de código aberto, disponível para uso de todos. Ele é distribuído sob os termos liberais da licença BSD modificada.

3.8 Splinter

O Splinter [20] é usado para escrever scripts de automação de navegador da web. O projeto tem dois objetivos principais:

- Fornece uma API comum e de alto nível em cima de ferramentas existentes de automação de navegador, como o Selenium. A API é uma camada de abstração amigável para humanos e projetada para facilitar a criação de scripts de forma eficiente.
- Fornecer funcionalidades integradas para lidar com pontos comuns de dor na automação de navegador.

Aqui estão alguns exemplos de como o Splinter pode ser usado:

- Testar aplicativos da web de forma automatizada.
- Rastrear dados da web de forma automatizada.
- Preencher formulários da web de forma automatizada.
- Clicar em links da web de forma automatizada.

4. Análise de Sistemas Existentes

Segundo o ONS [21] e seus sistemas, estes atuam na normatização da operação da distribuição da matriz energética no cenário nacional, além de cuidar da formatação destas normas e instruções operacionais e atuam principalmente nos cenários de pré-operação; operação em tempo real; e pós-operação.

A pré-operação consolida a programação eletroenergética diária através de intervenções nas restrições operativas dos reservatórios junto com o Controle Automático de Geração (CAG).

A operações em tempo real e tem como por objetivo coordenar, supervisionar e controlar o funcionamento operacional da Rede de Operação, além do devido funcionamento das instalações do SIN.

A pós-operação complementam a cadeia da Operação do Sistema com o apuramento dos dados das operações realizadas, bem como análises das ocorrências reportadas, perturbações energéticas e variações nas instalações elétricas do SIN, divulgações de resultados para os agentes do setor como órgãos governamentais, normativos, fiscalizadores e para a sociedade.

Todavia, este projeto tem como enfatizar um único item destes processos de atuação do ONS, que é o cenário de pós-operação relacionado a unificação das análises das ocorrências reportadas pelas UHEs através dos boletins diários via agrupamento de dados relacionais e API pública de consulta.

CAPÍTULO III

VISÃO DO SISTEMA

1. Riscos do Projeto

Tabela 1: Riscos do Projeto.

Nº	Descrição do Risco	Classificação
RI001	Complexidade na implementação dos dados relacionais referentes aos clusters, devido as alterações estruturais dos boletins gerados quanto sua padronização e nomenclaturas oficiais do ONS.	M
RI002	Complexidade na implementação de robôs nos quais utilizam frameworks <i>Web Scraping</i> para extração de dados dos boletins diários do ONS, devido a possibilidade de mecanismos <i>anti-crawling</i> utilizados como padrões de segurança cibernética que impedem que essas operações sejam realizadas por operações automatizadas.	A

2. Restrições do Projeto

Tabela 2: Restrições do Projeto.

Nº	Descrição da Restrição	Tipo
RP001	Implementações de modelos de aprendizagem de máquina.	Tecnológica
RP002	Operações com sistemas distribuídos para <i>Data Warehousing</i> .	Tecnológica
RP003	Autenticação e autorização por protocolos de segurança, como: <i>OAuth</i> , <i>Bearer Token</i> ou qualquer	Tecnológica

	outra tecnologia de validação de segurança para as consultas direcionadas ao SisBIN provido.	
R0P04	Técnicas e mecanismos para burlar funcionalidades de segurança durante as operações de extração de dados por robôs automatizados através de frameworks <i>Web Scraping</i> em Python.	Tecnológica

3. Regras do Negócio

Tabela 3: Regras de Negócio.

Código	Nome	Descrição
RN001	Manutenção dos boletins diários do ONS	Apenas o ONS é responsável pela manutenção permanente dos boletins diários gerados através do endereço eletrônico https://sdro.ons.org.br/SDRO/DIARIO/index.htm .
RN002	Persistência de dados obtidos através dos boletins diários	O sistema estabelece a persistência dos dados extraídos pelos relatórios diários de modo a organizar as informações e assim promover uma base uniforme de dados relacionais.
RN003	Atualização sob demanda da extração de dados remotos	Para estabelecer as devidas persistências das informações relevantes extraídos por neste sistema, os robôs automatizados serão acionados manualmente para geração de novos relatórios e atualização da base de dados local envolvida.
RN004	Imutabilidade das fontes ou origens das consultas remotas	O sistema não utilizará outras fontes para consulta de informações de interesse, apenas dentro do domínio do ONS, anteriormente citados na regra de negócio RN001. Sendo assim, a disponibilidade deste serviço é

		essencial para o devido funcionamento do sistema.
RN005	Padronização dos dados através de entidades relacionais	Os dados emitidos através dos relatórios dos robôs automatizados serão padronizados em formato de entidades relacionais, de modo a auxiliar no desenvolvimento de esquemas para a persistência na base de dados. Logo, promovendo a melhor compreensão de como os dados persistidos serão organizados e estruturados.
RN006	Acionamento via API dos robôs	O sistema não conterá mecanismos de acionamento dos robôs com autenticação ou autorização pelo serviço da API disponibilizada, pois este será para fins educacionais a fim de demonstrar o devido funcionamento de diversas funcionalidades automatizadas e integradas no sistema. Logo, o acionamento será feito através de um <i>endpoint</i> exposto de thread única.

4. Requisitos Funcionais

Tabela 4: Requisitos Funcionais.

Código	Nome	Descrição
RF001	API para consultas via endpoints	Este requisito aborda a criação de uma API para realização de consultas através de endpoints remotos aos usuários do sistema e assim facilitar a integração com demais sistemas adjacentes.
RF002	Base de dados relacional para configuração de grandes massas	Este requisito aborda a criação e configuração de uma base de dados relacional, na qual suporte a interpretação SQL para manipulação das estruturas de dados relacionais, a fim de persistir os dados gerados pelos robôs

	de dados dos boletins diários	automatizados e que serão disponibilizados nas operações de endpoints da API.
RF003	Robôs automatizados para extração de dados gerados pelos boletins do ONS	Este requisito aborda a criação de robôs que desempenhem o papel de extrair dados oriundos dos boletins gerados pelo ONS através de frameworks <i>Web Scraping</i> .
RF004	Operações de correlação de dados relacionais para geração de relatórios preditivos contra incidentes em UHEs	Este requisito aborda a criação de um endpoint específico para chamada de rotinas de cálculo científico e estatístico de modo a providenciar relatórios de predição contra possíveis incidentes em UHEs através de análise dos boletins diários do ONS de acordo com as normas da ANEEL.

5. Requisitos Não Funcionais

Tabela 5: Requisitos Não Funcionais.

Código	Nome	Descrição
RNF001	Arquitetura API REST	Este requisito aborda a utilização da arquitetura REST para a API desenvolvida.
RNF002	Interpretador de dados estruturais através do SQL	Este requisito aborda a utilização do padrão estrutural de Entidade-Relacionamento para manipulação de suas estruturas, persistidas na base de dados, através do SQL.

6. Mensagens do Sistema

Tabela 6: Mensagens do Sistema.

Código	Descrição
MSG001	EX_OK – Representa a finalização correta do sistema (código de saída “0”).
MSG002	EX_SOFTWARE – Representa uma finalização abrupta não tratada do sistema (código de saída “70”).

7. Lista de Casos de Uso

Tabela 7: Lista de Caso de Uso.

Código	Nome	Descrição
UC001	Acionar Bot de extração	Este caso de uso tem como finalidade a realização de operações de acionamento em escala de robô automatizado para extração de dados dos boletins diários do ONS via framework <i>Web Scraping</i> . Esta operação é decisiva para o processamento em cascata das demais operações citadas a seguir [UC002; UC003; UC004; UC005], se somente se, houver a necessidade do acionamento do robô.
UC002	Extrair dados do ONS	Este caso de uso tem como finalidade a realização de operações via framework <i>Web Scraping</i> para extração de arquivos de extensão CSV na plataforma Open Data ONS.
UC003	Processar dados extraídos via frameworks <i>Data Mining</i>	Este caso de uso tem como finalidade a realização de operações via frameworks <i>Data Mining</i> para tratamento dos dados

		extraídos após a leitura dos arquivos de extensão CSV da plataforma Open Data ONS.
UC004	Atualizar registros na base de dados	Este caso de uso tem como finalidade a realização de operações internas para atualização de todos os novos registros encontrados após devido tratamento dos dados processados.
UC005	Excluir arquivos temporários	Este caso de uso tem como finalidade a realização de operações internas para remoção de arquivos temporários que foram utilizados durante os procedimentos de processamento e atualização dos registros, em caso da existência de algum resquício.
UC006	Consultar predição via endpoint por período	Este caso de uso tem como finalidade a realização de operações de acionamento e consulta de resultados previstos via endpoint da API para obtenção de medidas estipuladas através de cálculos probabilísticos em análises de séries temporais utilizando modelo regressivo modular.
UC007	Extrair dados dos boletins publicados dos subsistemas	Este caso de uso tem como finalidade a realização de operações de consulta massiva na base de dados considerando todos os registros já publicados desde o ano 2000 para todas as regiões do SIN (Norte; Nordeste; Suldeste/Centro-Oeste; e Sul).

UC008	Realizar predição estocástica das séries temporais via framework Prophet	Este caso de uso tem como finalidade a realização de operações de cálculos probabilísticos através do framework Prophet e assim gerar resultados assertivos e práticos, considerando o período estipulado pelo usuário e todos os registros anteriores desde a última atualização do sistema, segundo o [UC004].
-------	--	--

8. Lista de Atores

Tabela 8: Lista de Atores.

Código	Nome	Descrição
001	Usuário	Tem acesso as análises preditivas de dados para predição de incidentes em UHEs fornecidos pelo sistema para todo o SIN e suas sub-regiões.

9. Diagrama Geral de Caso de Uso

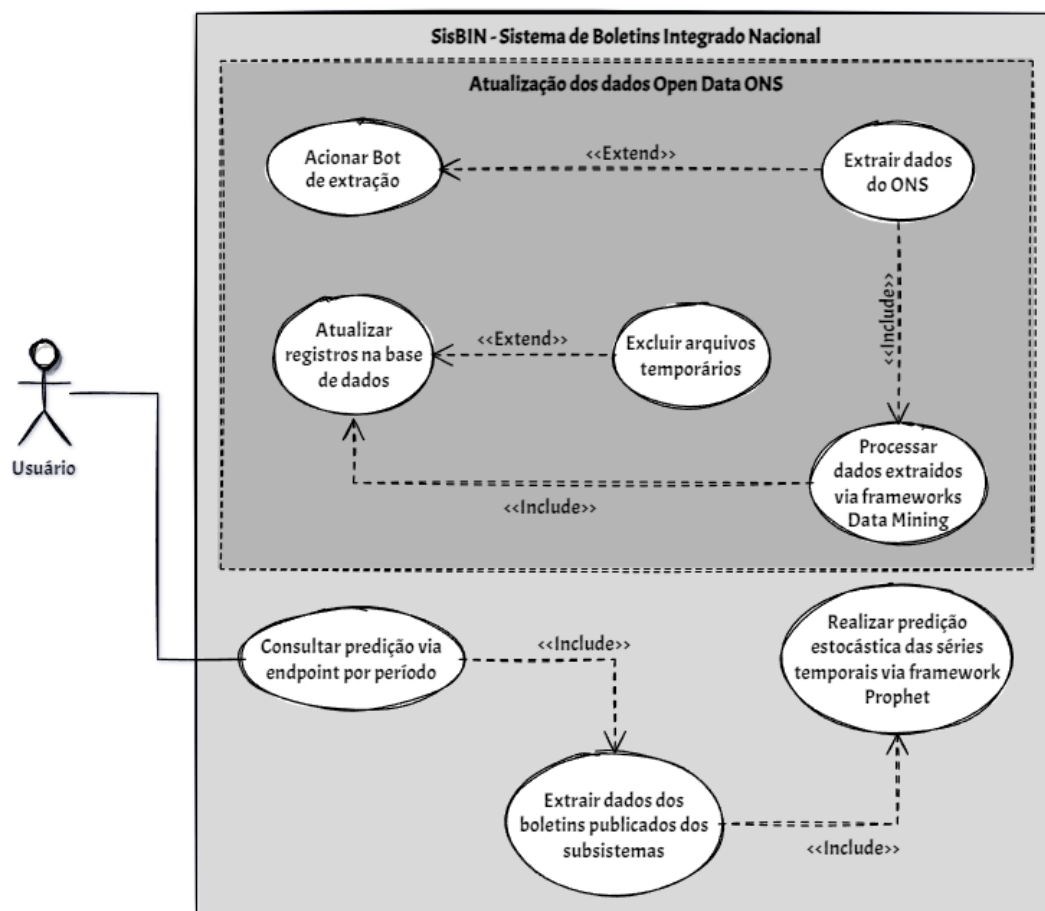


Figura 8: Diagrama de Caso de Uso do SisBIN para representação das operações diretas à API e afins.

10. Diagrama de Classe



Figura 9: Diagrama de Classes.

11. Diagrama de Sequência

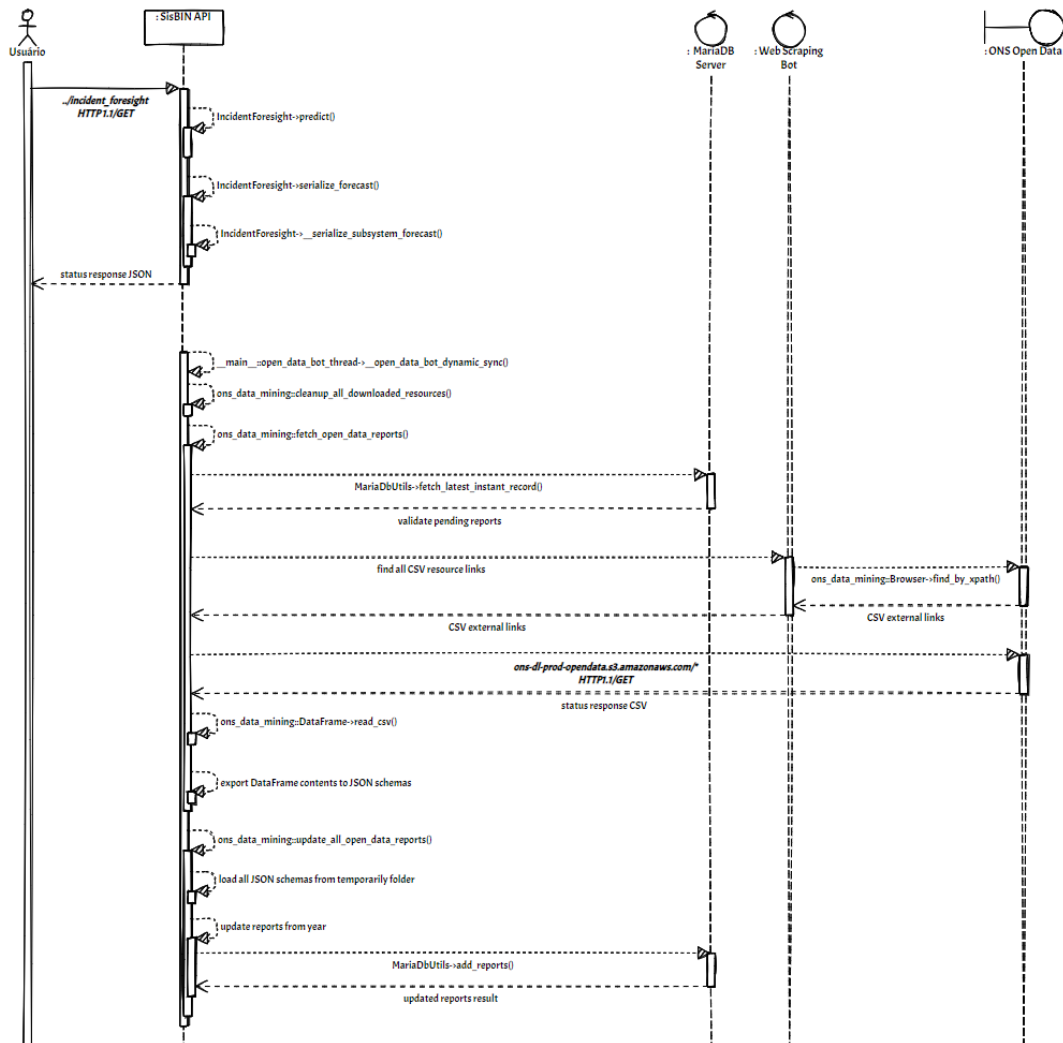


Figura 10: Diagrama de Sequência.

CAPÍTULO IV

ESPECIFICAÇÃO DE CASO DE USO

UC001 – Acionar Bot de extração

1. Descrição

Este caso de uso tem como finalidade a realização de operações de acionamento em escala de robô automatizado para extração de dados dos boletins diários do ONS via framework *Web Scraping*. Esta operação é decisiva para o processamento em cascata das demais operações citadas a seguir [UC002; UC003; UC004; UC005], se somente se, houver a necessidade do acionamento do robô.

2. Pré-condições

- O sistema precisa ter configurado a base de dados local, segundo as restrições pré-definidas [RE001; RE002; RE003; e RE004];
- O sistema precisa ter acesso ao WebDriver especificado nas configurações do arquivo local YAML;
- O sistema precisa ter acesso aos servidores Open Data ONS, sem restrições de firewall ou bloqueios de comunicação através das portas de conexão padrão 80 e 443, HTTP e HTTPS respectivamente [RI001; e RI002].

3. Fluxos

Tabela 9: fluxo principal referente ao caso de uso UC001 - Acionar Bot de extração.

Código	Fluxo	Descrição
UC001-FP1	Principal	O sistema é iniciado e todas as configurações locais foram devidamente processadas. [RN004; e RNF001]
UC001-FP2	Principal	Uma sub-rotina é criada com o intuito de verificar a integridade dos dados já coletados. [RN005; e RN006]

UC001-FP3	Principal	É verificado se existe a necessidade de atualização dos registros.
UC001-FP4	Principal	O caso de uso se encerra.

Tabela 10: fluxo alternativo referente ao caso de uso UC001 - Acionar Bot de extração.

Código	Fluxo	Descrição
UC001-FA1	Alternativo	É identificado a necessidade de atualização dos registros [UC001-FP3], o bot é acionado para realização de varredura nos arquivos remotos Open Data ONS. [RF002]
UC001-FA2	Alternativo	Os dados são extraídos [UC002].

Tabela 11: fluxo de exceção referente ao caso de uso UC001 - Acionar Bot de extração.

Código	Fluxo	Descrição
UC001-FE1	Exceção	Não é possível realizar operações de acionamento do bot devido a falha no carregamento do WebDriver e a mensagem [MSG002] é exibida no console da aplicação.
UC001-FE2	Exceção	O sistema é finalizado.

4. Pós-condições

- O bot deverá ser devidamente acionado para o prosseguimento das operações em [UC002].

5. Pontos de Extensão

- UC002 – Extrair dados do ONS.

6. Pontos de Inclusão

- N/A.

UC002 – Extrair dados do ONS

1. Descrição

Este caso de uso tem como finalidade a realização de operações via framework Web Scraping para extração de arquivos de extensão CSV na plataforma Open Data do ONS.

2. Pré-condições

- O sistema precisa ter concluído com êxito as operações definidas em [UC001].

3. Fluxos

Tabela 12: fluxo principal referente ao caso de uso UC002 – Extrair dados do ONS.

Código	Fluxo	Descrição
UC002-FP1	Principal	O bot é acionado para realizar as operações de busca na página do Open Data ONS. [RN001; RN004; RF003]
UC002-FP2	Principal	Após a consulta, é retornado todos os possíveis endereços eletrônicos que possuem relação aos dados em formato CSV. [RN005]
UC002-FP3	Principal	Após análise, os arquivos em formato CSV são baixados localmente em um diretório temporário, nos quais serão posteriormente processados em [UC003].
UC002-FP4	Principal	O caso de uso se encerra.

Tabela 13: fluxo de exceção referente ao caso de uso UC002 – Extrair dados do ONS.

Código	Fluxo	Descrição
UC002-FE1	Exceção	Não é possível realizar operações de consultas nos servidores Open Data ONS e a mensagem [MSG002] é exibida no console da aplicação.
UC002-FE2	Exceção	O sistema é finalizado.

4. Pós-condições

- O bot deverá ser devidamente acionado para o prosseguimento das operações em [UC003].

5. Pontos de Extensão

- N/A.

6. Pontos de Inclusão

- UC003 – Processar dados extraídos via frameworks Data Mining.

UC003 – Processar dados extraídos via frameworks *Data Mining*

1. Descrição

Este caso de uso tem como finalidade a realização de operações via frameworks *Data Mining* para tratamento dos dados extraídos após a leitura dos arquivos de extensão CSV da plataforma Open Data do ONS.

2. Pré-condições

- O sistema precisa ter concluído com êxito as operações definidas em [UC002].

3. Fluxos

Tabela 14: fluxo principal referente ao caso de uso UC003 – Processar dados extraídos via frameworks *Data Mining*.

Código	Fluxo	Descrição
UC003-FP1	Principal	Após as operações de extração, é iniciado operações de processamento em massa dos dados nos quais foram consultados e extraídos do Open Data ONS. [RN003]
UC003-FP2	Principal	Frameworks de <i>Data Mining</i> são acionados para realizar a devida conversão dos dados solicitados para modelos de <i>dataset</i> . [RN005; RF004]
UC003-FP3	Principal	Após processamento, os arquivos em formato CSV são devidamente convertidos e manipulados em dados nos quais serão trabalhados em formato JSON, estes serão exportados e acurados em [UC004].
UC003-FP4	Principal	O caso de uso se encerra.

Tabela 15: fluxo de exceção referente ao caso de uso UC003 – Processar dados extraídos via frameworks *Data Mining*.

Código	Fluxo	Descrição
UC003-FE1	Exceção	Não é possível realizar operações de processamento devido inconsistência dos dados apresentados após realizar consultas nos servidores Open Data ONS e a mensagem [MSG002] é exibida no console da aplicação.
UC003-FE2	Exceção	O sistema é finalizado.

4. Pós-condições

- O sistema deverá ter consigo os dados processados para o prosseguimento das operações em [UC004].

5. Pontos de Extensão

- N/A.

6. Pontos de Inclusão

- UC004 – Atualizar registros na base de dados.

UC004 – Atualizar registros na base de dados

1. Descrição

Este caso de uso tem como finalidade a realização de operações internas para atualização de todos os novos registros encontrados após devido tratamento dos dados processados.

2. Pré-condições

- O sistema precisa ter concluído com êxito as operações definidas em [UC003].

3. Fluxos

Tabela 16: fluxo principal referente ao caso de uso UC004 - Atualizar registros na base de dados.

Código	Fluxo	Descrição
UC004-FP1	Principal	Após as operações de processamento, é iniciado operações de atualização dos registros em massa nos quais foram processados, consultados e extraídos do Open Data ONS. [RN002; RF002; RNF002]
UC004-FP2	Principal	O caso de uso se encerra.

Tabela 17: fluxo de exceção referente ao caso de uso UC004 - Atualizar registros na base de dados.

Código	Fluxo	Descrição
UC004-FE1	Exceção	Não é possível realizar operações de atualização dos dados processados devido inconsistência dos dados apresentados após realizar consultas nos servidores Open Data ONS e a mensagem [MSG002] é exibida no console da aplicação.
UC004-FE2	Exceção	O sistema é finalizado.

4. Pós-condições

- N/A.

5. Pontos de Extensão

- UC005 – Excluir arquivos temporários.

6. Pontos de Inclusão

- N/A.

UC005 – Excluir arquivos temporários

1. Descrição

Este caso de uso tem como finalidade a realização de operações internas para remoção de arquivos temporários que foram utilizados durante os procedimentos de processamento e atualização dos registros, em caso da existência de algum resquício.

2. Pré-condições

- O sistema precisa ter concluído com êxito as operações definidas em [UC004].

3. Fluxos

Tabela 18: fluxo principal referente ao caso de uso UC005 – Excluir arquivos temporários.

Código	Fluxo	Descrição
UC005-FP1	Principal	Após as operações de atualização, é iniciado operações de varredura nos registros nos quais foram atualizados através de leitura local de arquivos temporários.
UC005-FP2	Principal	Os arquivos temporários são devidamente removidos do diretório temporário criado pelo sistema.
UC005-FP3	Principal	O caso de uso se encerra.

4. Pós-condições

- N/A.

5. Pontos de Extensão

- N/A.

6. Pontos de Inclusão

- N/A.

UC006 – Consultar predição via endpoint por período

1. Descrição

Este caso de uso tem como finalidade a realização de operações de acionamento e consulta de resultados previstos via endpoint da API para obtenção de medidas estipuladas através de cálculos probabilísticos em análises de séries temporais utilizando modelo regressivo modular.

2. Pré-condições

- O sistema precisa ter concluído com êxito as operações definidas em [UC004] para devida obtenção de registros históricos;
- O ator precisa informar o período inicial no qual será consultado, o valor padrão é a data atual da consulta;
- O ator precisa informar o período final no qual será consultado, o valor padrão é de 7 dias após a data atual da consulta, em caso de não declaração; e
- O sistema precisa garantir que o período estipulado está de acordo com dados gerados pelos boletins ONS no contexto histórico.

3. Fluxos

Tabela 19: fluxo principal referente ao caso de uso UC006 – Consultar predição via endpoint por período.

Código	Fluxo	Descrição
UC006-FP1	Principal	O ator informa ao sistema o período estipulado no qual será gerado os relatórios preditivos das regiões do SIN. [RF001]
UC006-FP2	Principal	É iniciado a preparação de operações nas quais serão configurados o modelo preditivo, os dados relevantes para as previsões e pré-processamento dos dados gerados ao endpoint levando em conta o período proposto pelo ator. [RF004]
UC006-FP3	Principal	O caso de uso se encerra.

4. Pós-condições

- É necessário a existência de registros histórico para cada subsistema do SIN dentro da base de dados local do sistema.

5. Pontos de Extensão

- N/A.

6. Pontos de Inclusão

- UC007 – Extrair dados dos boletins publicados dos subsistemas.

UC007 – Extrair dados dos boletins publicados dos subsistemas

1. Descrição

Este caso de uso tem como finalidade a realização de operações de consulta massiva na base de dados considerando todos os registros já publicados desde o ano 2000 para todas as regiões do SIN (Norte; Nordeste; Suldeste/Centro-Oeste; e Sul).

2. Pré-condições

- O sistema precisa ter concluído com êxito as operações definidas em [UC006].

3. Fluxos

Tabela 20: fluxo principal referente ao caso de uso UC007 – Extrair dados dos boletins publicados dos subsistemas.

Código	Fluxo	Descrição
UC007-FP1	Principal	Após a preparação do modelo preditivo, o sistema realiza consultas massivas para extração e processamento de todos os registros publicados para cada subsistema. [RF002]
UC007-FP2	Principal	O caso de uso se encerra.

4. Pós-condições

- É necessário a preparação dos dados nos quais serão correlacionados dentro do modelo preditivo dentro do período medido pelo ator.

5. Pontos de Extensão

- N/A.

6. Pontos de Inclusão

- UC008 – Realizar predição estocástica das séries temporais via framework Prophet.

UC008 – Realizar predição estocástica das séries temporais via framework Prophet

1. Descrição

Este caso de uso tem como finalidade a realização de operações de cálculos probabilísticos através do framework Prophet e assim gerar resultados assertivos e práticos, considerando o período estipulado pelo usuário e todos os registros anteriores desde a última atualização do sistema segundo o [UC004].

2. Pré-condições

- O sistema precisa ter concluído com êxito as operações definidas em [UC004; e UC007].

3. Fluxos

Tabela 21: fluxo principal referente ao caso de uso UC008 – Realizar predição estocástica das séries temporais via framework Prophet.

Código	Fluxo	Descrição
UC008-FP1	Principal	Após a preparação do modelo preditivo, o sistema realiza a modelagem dos dados que foram obtidos pelas consultas massivas de todos os registros publicados para cada subsistema. [RF004]
UC008-FP2	Principal	Após a modelagem do modelo preditivo, o sistema realiza a preparação de previsões dentro do período estipulado pelo ator. [RF004]
UC008-FP3	Principal	Após a preparação de previsões, o sistema realiza a criação de módulos contextuais (ds^2 ; y^3 ; $yhat^4$; $yhat_lower^5$; e

² Notação que representa o instante de tempo do valor previsto durante a análise preditiva.

³ Notação que representa o valor real que será utilizado nas operações de predição.

⁴ Notação que representa o valor previsto gerado pelo modelo preditivo.

⁵ Notação que representa o valor mínimo previsto gerado pelo modelo preditivo.

		<i>yhat_higher</i> ⁶) referente as previsões criadas pelo framework Prophet, nos quais serão decisivas para formatação dos dados em formato JSON.
UC008-FP4	Principal	O sistema realiza a serialização dos dados levando em conta os módulos contextuais preditivos citados em [UC008-FP3].
UC008-FP5	Principal	O sistema retorna através do endpoint consultado pelo ator definido em [UC006] dentro do período proposto com as previsões energéticas dos subsistemas do SIN.
UC008-FP6	Principal	O caso de uso se encerra.

4. Pós-condições

- N/A.

5. Pontos de Extensão

- N/A.

6. Pontos de Inclusão

- N/A.

⁶ Notação que representa o valor máximo previsto gerado pelo modelo preditivo.

CAPÍTULO V

MODELAGEM DE DADOS

[Este capítulo deverá propor ao aluno aplicar todo o conhecimento adquirido sobre modelagem de dados]

1. Diagrama de Entidade Relacional

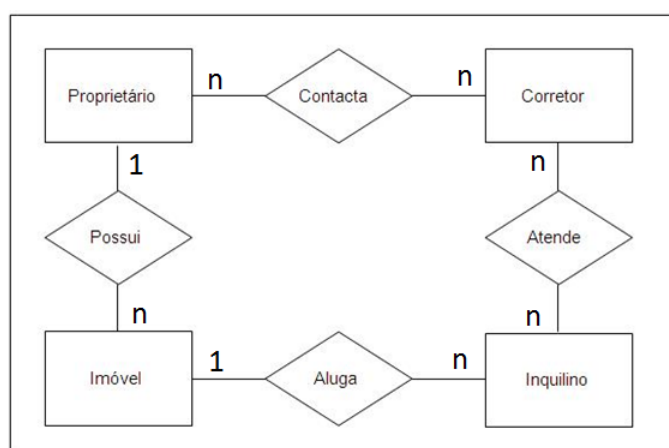


Figura 06 – Diagrama de Entidade Relacional

2. Diagrama de Entidade Relacional

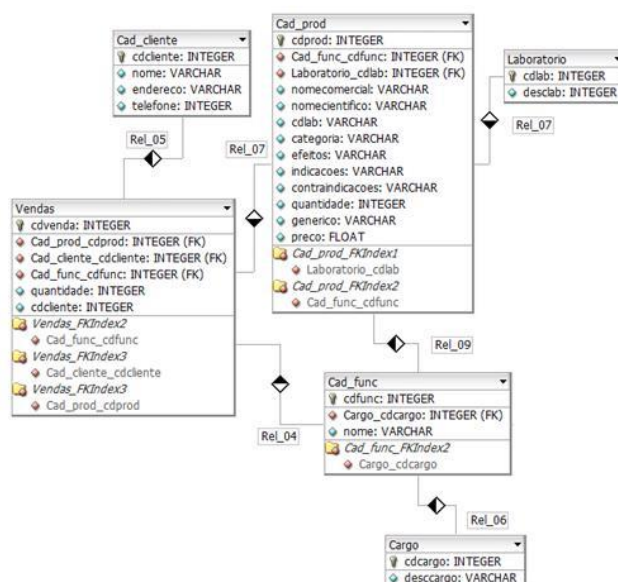


Figura 07 – Diagrama de Banco de Dados: MER

3.1 Dicionário de Dados

3.1.1 Tabela [Nome da Tabela]

Campo	Tipo/Tamanho	Obrigatório	Comentário
[Informar o nome do campo]	[Informar o tipo do campo]	[Informar S para obrigatório e N para não obrigatório]	[Comentar sobre o campo]
RELACIONAMENTOS			
Tabela	Descrição		
[Informar o nome da tabela]	[Realizar uma descrição sobre o campo da tabela que se deseja relacionar]		

[Padronizar o nome das tabelas. Exemplo: DBProcesso]

1.1 Campos da Tela

Item	Nome do Campo	Tipo	Tam	Máscara	Obrigatório	Valor Padrão	Editável	Domínio	Visível
001	Cliente	Alfanumérico	100	N/A	N	N/A	S	Nome do campo na base de dados	S

1.2 Comandos da Tela

Item	Comando	Ação	Restrições/Observações
01	Gravar	Grava o processo no sistema	Todos os campos obrigatórios devem ser preenchidos

[Sempre alterar entre imagem do prototipo e a descrição da tela]

CONCLUSÃO

[Informar as considerações finais relacionado ao trabalho]

REFERÊNCIAS

1. ONS - OPERADOR NACIONAL DO SISTEMA ELÉTRICO. O que é ONS. **ONS - Operador Nacional do Sistema Elétrico**. Disponível em: <<https://www.ons.org.br/paginas/sobre-o-ons/o-que-e-ons>>. Acesso em: 02 Setembro 2021.
2. HICKEY, A. M.; DAVIS, A. M. **Requirements Elicitation and Elicitation Technique Selection**: Model for Two Knowledge - Intensive Software Development Processes. 36ª Conferência Internacional Anual do Havaí. Havaí: [s.n.]. 2003.
3. SHERMAN, R. Chapter 18 - Project Management. In: SHERMAN, R. **Business Intelligence Guidebook**: From Data Integration to Analytics. [S.l.]: Elsevier Inc., 2015. p. 449-492.
4. UNIVERSIDADE FEDERAL DE SANTA CATARINA - UFSC. REtraining - Guia facetado de Técnicas de Elicitação de Requisitos. Disponível em: <<https://retraining.inf.ufsc.br/guia/app/classificacoes/tecnicas-de-elicacao-de-requisitos/entidades/tecnicas-de-elicacao-de-requisitos-brainstorming>>. Acesso em: 17 Outubro 2021.
5. SELENIUM PROJECT. WebDriver | Selenium. Disponível em: <<https://www.selenium.dev/documentation/webdriver/>>. Acesso em: 07 Novembro 2021.
6. W3C. World Wide Web Consortium, 05 Junho 2018. Disponível em: <<https://www.w3.org/TR/webdriver1/>>. Acesso em: 07 Novembro 2021.
7. PYTHON SOFTWARE FOUNDATION. Welcome to Python.org. Disponível em: <<https://www.python.org>>. Acesso em: 07 Novembro 2021.
8. PYTHON SOFTWARE FOUNDATION. Extending and Embedding the Python Interpreter. Disponível em: <<https://docs.python.org/3/extending/index.html#extending-index>>. Acesso em: 07 Novembro 2021.
9. PYTHON SOFTWARE FOUNDATION. Python/C API Reference Manual. Disponível em: <<https://docs.python.org/3/c-api/index.html#c-api-index>>. Acesso em: 07 Novembro 2021.
10. MARIADB FOUNDATION. About MariaDB Server - MariaDB.org. Disponível em: <<https://mariadb.org/about/>>. Acesso em: 08 Novembro 2021.
11. RAMÍREZ, S. FastAPI framework, high performance, easy to learn, fast to code, ready for production. Disponível em: <<https://fastapi.tiangolo.com>>. Acesso em: 09 Novembro 2021.
12. THE LINUX FOUNDATION. OpenAPI - The world's most widely used API description standard. Disponível em: <<https://www.openapis.org>>. Acesso em: 09 Novembro 2021.

13. JSON SCHEMA. JSON Schema. Disponível em: <<https://json-schema.org>>. Acesso em: 09 Novembro 2021.
14. ENCODE OSS. Uvicorn - An ASGI web server, for Python. Disponível em: <<https://www.uvicorn.org>>. Acesso em: 09 Novembro 2021.
15. MAGICSTACK. MagicStack/uvloop - Ultra fast asyncio event loop. Disponível em: <<https://github.com/MagicStack/uvloop>>. Acesso em: 09 Novembro 2021.
16. MAGICSTACK. MagicStack/httptools - Fast HTTP parser. Disponível em: <<https://github.com/MagicStack/httptools>>. Acesso em: 09 Novembro 2021.
17. FACEBOOK OPEN SOURCE. Prophet - Forecasting at scale. Disponível em: <<https://facebook.github.io/prophet/>>. Acesso em: 14 Maio 2024.
18. TAYLOR, S. J.; LETHAM, B. Forecasting at Scale. **PeerJ Publishing**, 27 Setembro 2017.
19. NUMPY. NumPy - About Us. Disponível em: <<https://numpy.org>>. Acesso em: 14 Maio 2024.
20. COBRA TEAM. Why Use Splinter? - Splinter 0.21.0 documentation. Disponível em: <<https://splinter.readthedocs.io/en/latest/why.html>>. Acesso em: 14 Maio 2024.
21. ONS. ONS - Atuação. Disponível em: <<https://www.ons.org.br/paginas/sobre-ons/atuacao>>. Acesso em: 15 Maio 2024.