

## **Sistema de Monitoramento e Supervisão de Unidades Transformadoras e Subestação de Energia <sup>1</sup>**

**João Gabriel Foletto Schefer<sup>2</sup>, Guilherme Eckhardt<sup>3</sup>, Luis Gustavo Tabile<sup>4</sup>, Manuel Osorio Binelo<sup>5</sup>, Paulo Sérgio Sausen<sup>6</sup>, Luis Fernando Sauthier<sup>7</sup>**

<sup>1</sup> Pesquisa institucional desenvolvida no Departamento de Ciências Exatas e Engenharias (DCEeng), pertence ao Grupos de Automação e Controle Industrial (GAIC), projeto financiado pela Companhia Estadual de Distribuição de Energia Elétrica (CEEE-D).

<sup>2</sup> Bolsista PROFAP, Aluno do Curso de Ciência da Computação da Unijuí.

<sup>3</sup> Bolsista PROFAP, Aluno do Curso de Ciência da Computação da Unijuí.

<sup>4</sup> Professor Pesquisador do Grupo de Matemática Aplicada e Computacional - GMAC - Unijuí

<sup>5</sup> Professor Pesquisador do Grupo de Automação Industrial e Controle - GAIC - Unijuí.

<sup>6</sup> Professor Pesquisador do Grupo de Automação Industrial e Controle - GAIC - Unijuí.

<sup>7</sup> Professor Pesquisador do Grupo de Automação Industrial e Controle - GAIC - Unijuí.

<sup>8</sup> Professor Pesquisador do Grupo de Automação Industrial e Controle - GAIC - Unijuí.

### **INTRODUÇÃO**

O presente projeto visa a criação de um sistema avançado de monitoramento de subestações de energia elétrica. O sistema consiste em placas eletrônicas dotadas de vários sensores que monitoram grandezas como tensão elétrica, corrente elétrica, temperatura, pressão, entrada da subestação, entre outros. Esse sistema registrará os dados na nuvem e disponibilizará os valores monitorados via internet. Neste contexto, o propósito central deste artigo é fornecer uma visão abrangente dos resultados alcançados até o momento atual. Serão abordados os progressos realizados tanto no desenvolvimento do frontend quanto no backend do sistema de monitoramento em operação.

### **METODOLOGIA**

Esse projeto ainda encontra-se em desenvolvimento, porém grande parte dele já foi concluída, como o backend e frontend. O backend do projeto foi feito utilizando a linguagem de programação Python, pois ela é uma linguagem de alto nível orientada à objeto, de tipagem dinâmica e forte, interpretada e interativa (BORGES, Luiz Eduardo). A linguagem Python destaca-se por sua sintaxe clara e concisa, o que contribui significativamente para a legibilidade do código-fonte e, por consequência, aumenta a produtividade durante o desenvolvimento.

Essa linguagem oferece uma variedade de estruturas de alto nível, como listas, dicionários, manipulação de data/hora, números complexos e outros recursos. Além disso,

Python conta com uma vasta biblioteca de módulos pré-implementados, prontos para serem utilizados. Adicionalmente, é possível integrar frameworks de terceiros, ampliando ainda mais as funcionalidades disponíveis.

Para o backend, um framework foi utilizado nesse projeto, o Django Framework. Além de viabilizar a reutilização de códigos, simplificando consideravelmente o processo de desenvolvimento, o framework disponibiliza também uma interface administrativa que torna mais conveniente a administração do conteúdo. Uma das maiores vantagens do Django é a grande documentação disponível, garantindo uma experiência mais completa e esclarecedora.

Na parte do frontend foi utilizado um kit de desenvolvimento, o Flutter. Esse kit de desenvolvimento é fundamentado em Dart, o qual viabiliza a concepção de aplicativos compilados de forma nativa para múltiplas plataformas, Android, iOS, Windows, Mac, Linux, Fuchsia e Web.

Por fim, para banco de dados, foi utilizado o PostgreSQL. Ele é uma ótima opção para aplicações que necessitam escalabilidade e um alto nível de disponibilidade e processa um conjunto de dados muito rapidamente.

## **RESULTADOS E DISCUSSÃO**

O desenvolvimento do projeto resultou em avanços significativos em termos de funcionalidade e desempenho. A integração de Python com o framework Django no backend estabeleceu uma base sólida e eficaz para a lógica de negócios, permitindo a organização eficiente de modelos de dados, operações de banco de dados e rotas de aplicativo. A combinação de Python e Django ofereceu uma sinergia robusta, agilizando o desenvolvimento e a manutenção do projeto por meio da arquitetura MVC, que promoveu a reutilização de código e escalabilidade.

A implementação do Flutter no frontend resultou em melhorias notáveis na experiência do usuário e na interface do sistema. A capacidade de compilar nativamente para várias plataformas ampliou o alcance e a acessibilidade do projeto. A interface se destacou por sua modernidade, responsividade e interatividade, atendendo às expectativas de usabilidade e estética. A adoção do Flutter proporcionou coesão visual e funcional, simplificando o desenvolvimento para múltiplas plataformas e resultando em uma interface de usuário atrativa e consistente.

Logo ao acessar o sistema, podemos ver um painel com algumas informações sobre as placas (Figura 1).

Figura 1 - Tela inicial do sistema apresentando dados sobre os dispositivos.

Dispositivos  Sensores  resetar alarme

Timestamp	Disp.	V-AS (V)	V-BS (V)	V-CS (V)	I-AS (A)	I-BS (A)	I-CS (A)	Patv-AS (kW)	Patv-BS (kW)	Patv-CS (kW)	Papv-AS (kVA)	Papv-BS (kVA)	Papv-CS (kVA)	Pre-AS (kVar)	Pre-BS (kVar)	Pre-CS (kVar)	FP-AS	FP-BS	FP-CS	
14/08/23 09:37:37	SUBS00	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14/08/23 09:37:37	SUBS01	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14/08/23 09:37:37	SUBS02	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14/08/23 09:37:37	SUBS03	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14/08/23 09:37:37	SUBS04	225.96	222.67	225.4	0.15	0.05	0.21	0.0	0.0	0.0	0.035	0.012	0.049	0.035	0.0	0.049	0.02	0.01	0.01	0.01
14/08/23 09:37:37	SUBS05	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14/08/23 09:37:37	SUBS06	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Na parte superior esquerda, pode-se observar as palavras “Dispositivos” e “Sensores”. Se clicarmos em dispositivos, vamos selecionar quais dispositivos gostaríamos de ver informações. O mesmo acontece em “Sensores”, Se clicarmos ali, podemos selecionar os sensores que estão em tal dispositivo (Figura 2 e Figura 3).

Figura 2 - Tela de filtragem de dispositivos.

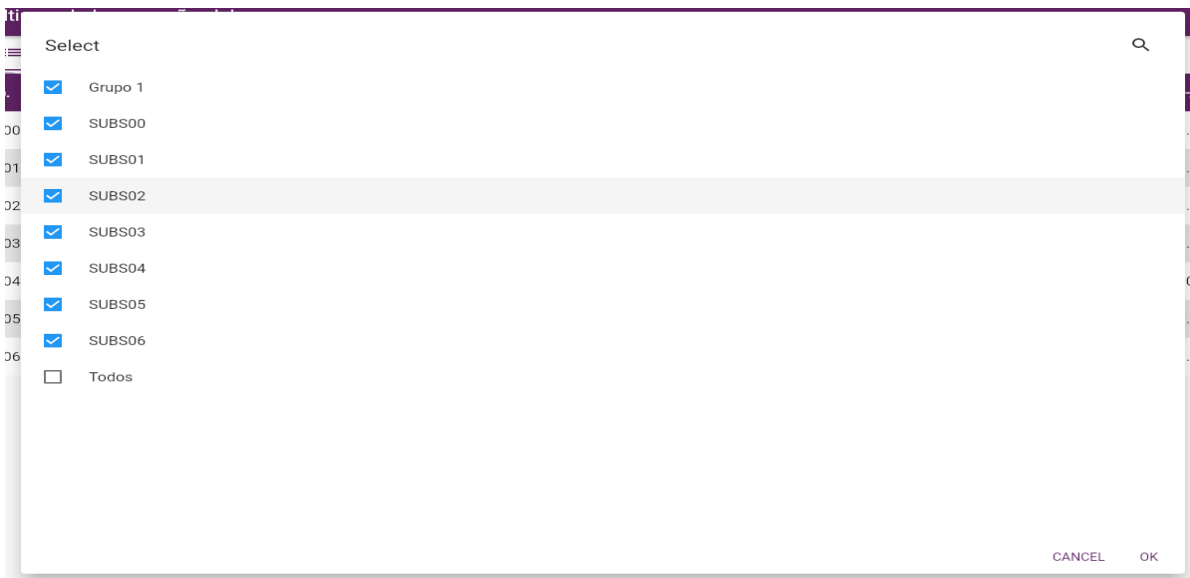
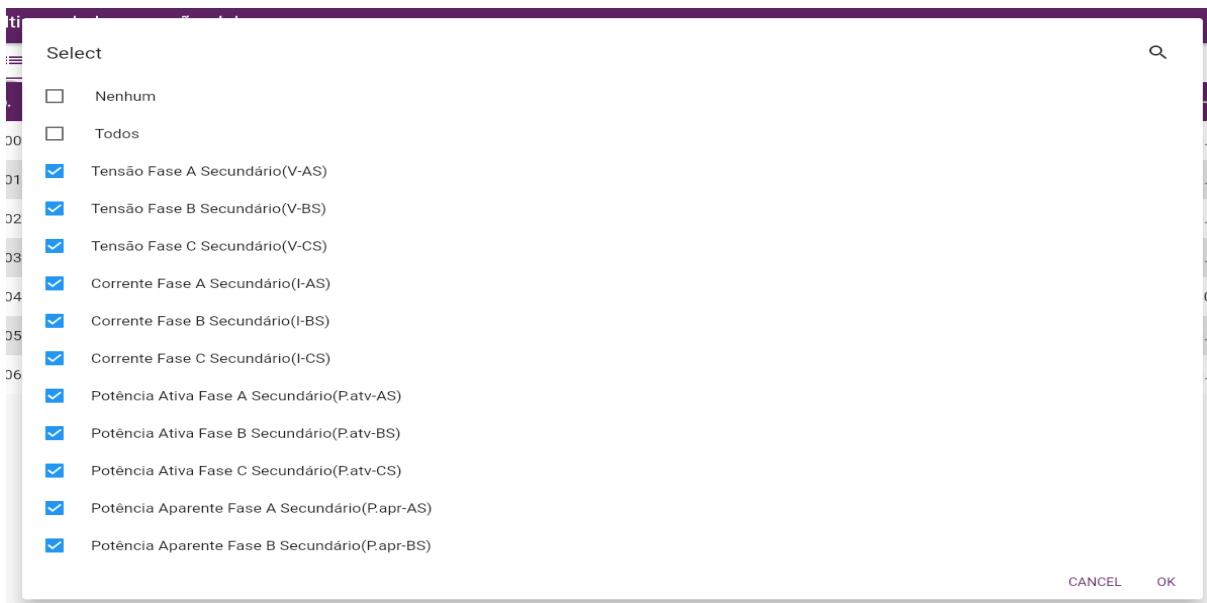
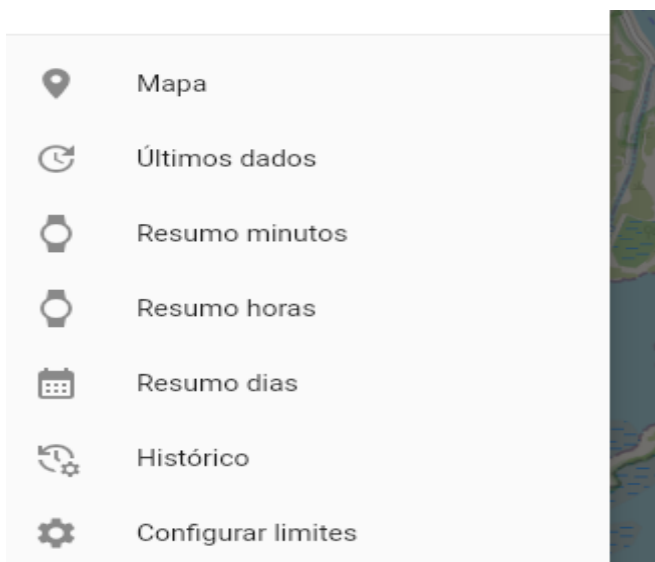


Figura 3 - Tela de filtragem de sensores.



Na parte superior esquerda temos uma opção para clicar e podermos navegar por outras partes do sistema, como mostra a Figura 4.

Figura 4 - Interface de navegação da aplicação através do menu lateral.



O uso do PostgreSQL como banco de dados foi crucial para assegurar a segurança, confiabilidade e escalabilidade do sistema. Sua eficácia na gestão de dados, incluindo operações complexas, preservou a integridade das informações. A escolha do PostgreSQL garantiu confiabilidade e eficiência nas operações de armazenamento e recuperação de dados, facilitada pela compatibilidade com o framework Django para integração com o backend.

## CONSIDERAÇÕES FINAIS

Em geral, o projeto revelou resultados sólidos ao unir de forma eficaz Python, Django, Flutter e PostgreSQL. A integração do Django no backend proporcionou base robusta para a lógica de negócios, enquanto o Flutter no frontend melhorou a experiência do usuário com interfaces modernas e coesas. O PostgreSQL trouxe segurança e escalabilidade ao sistema. Essa sinergia tecnológica destaca a importância da escolha criteriosa das ferramentas para criação de soluções eficientes e inovadoras, com potencial para enfrentar desafios em constante evolução.

**Palavras-chave:** Python. Django. Flutter. PostgreSQL.

## AGRADECIMENTOS

Os autores expressam seu agradecimento à UNIJUÍ e à CEEE-D pelo apoio concedido por meio da bolsa PROFAP-GR. Essa bolsa proporcionou oportunidades valiosas para aplicar nosso aprendizado por meio da idealização deste projeto. Além disso, agradecemos ao Grupo de Automação Industrial e Controle (GAIC) pela disponibilização do espaço para o desenvolvimento deste sistema.

## REFERÊNCIAS BIBLIOGRÁFICAS

BORGES, Luiz Eduardo.

Conforme normas da ABNT. Python para Desenvolvedores: Aborda Python 3.3. São Paulo, SP - Brasil: Rubens Prates, Outubro de 2014.

DAUZON, Samuel. BENDORAITIS, Aidas. RAVINDRAN, Arun. Django: Web Development with Python. Birmingham B3 2PB, UK: Packt Publishing Ltd, August 2016.

MILANI, André. PostgreSQL Guia do Programador. São Paulo, SP - Brasil: Novatec Editora Ltda, Abril /2008.